TECHNICAL RESEARCH REPORT

Recovering Information from Summary Data

by C. Faloutsos, H.V. Jagadish,
N.D. Sidiropoulos

T.R. 97-7

Sponsored by
the National Science Foundation
Engineering Research Center Program,
the University of Maryland,
Harvard University,
and Industry

Recovering Information from Summary Data

*Christos Faloutsos**
Dept. of Computer Science and
Inst. of Systems Research
Univ. of Maryland
College Park MD 20742
christos@cs.umd.edu

H. V. Jagadish
AT&T Laboratories
Murray Hill, NJ 07974
jag@research.att.com

N. D. Sidiropoulos
Inst. for Systems Research
University of Maryland
College Park MD 20742
nikos@glue.umd.edu

October 28, 1996

Abstract

Data is often stored in summarized form, as a histogram of aggregates (COUNTs, SUMs, or AVeraGes) over specified ranges. Queries regarding specific values, or ranges different from those stored, cannot be answered exactly from the summarized data. In this paper we study how to estimate the original detail data from the stored summary.

We formulate this task as an *inverse problem*, specifying a well-defined cost function that has to be optimized under constraints. In particular, we propose the use of a *Linear Regularization* method, which "maximizes the smoothness" of the estimate. Our main theoretical contribution is a Theorem, which shows that, for "smooth" enough distributions, we can achieve *full recovery* from summary data. Our theorem is closely related to the well known Shannon-Nyquist sampling theorem.

We describe how to apply this theory to a variety of database problems that involve partial information, such as OLAP, data warehousing and histograms in query optimization. Our main practical contribution is that the Linear Regularization method is extremely effective, both on synthetic and on real data. Our experiments show that the proposed approach almost consistently outperforms the "uniformity" assumption, achieving significant savings in root-mean-square error: up to 20% for stock price data, and up to 90% for smoother data sets.

*This research was partially funded by the National Science Foundation under Grants No. EEC-94-02384, IRI-9205273 and IRI-9625428.

1 Introduction

Consider the problem of an unknown set of numbers x_i $(i = 1, \ldots N)$, for which we are given some partial information. For example, x_i could represent the total sales for the i-th day, and we could be given only the monthly total sales. Suppose that we also have some additional, a-priori information, for example, that the sales patterns are "smooth", without abrupt jumps (*i.e.*, $x_i \approx x_{i+1}$). The goal is to recover the unknown values as best as we can[1].

In a multi-dimensional setting, this problem becomes even more interesting. Suppose that the unknown numbers are the counts $c_{i,j}$ of employees of a company, for each age-bracket i and for each salary-bracket j; suppose that we are only given the age- and salary-histograms, that is the counts $c_{i,*}$ for the i-th age-bracket and the counts $c_{*,j}$ for the j-th salary-bracket. The goal is to estimate the unknown $c_{i,j}$ counts.

This sort of problem arises in a host of different situations. Data is summarized over discrete ranges to create a database of manageable size for storage, manipulation, and display. Often, there is a need to respond to queries that can be answered accurately only from the base data, but that must be answered quickly from the summarized data. The task then is to reconstruct as good an estimate of the original base data as possible. Applications of such a generic reconstruction method abound:

- Query optimization: DBMSs typically maintain histograms [15] reporting the number of tuples for selected attribute-value ranges. Queries may select only specific values, or select ranges that only partially overlap with the value ranges used in the histogram. Cost estimation for such queries will benefit from an accurate reconstruction of attribute-value occurrences for the queried value(-range). Similarly, range queries on multiple attributes will benefit from an accurate synthesis and extrapolation from the histograms of value distributions for individual attributes.

- Data warehousing [30]: The idea is that the central site will have meta-data, and condensed information (*e.g.*, summary data) from each participating site, which has detailed information. Accessing the remote site might be slow and/or expensive; a cheap, accurate estimate of the missing information is attractive.

- Transaction recording systems: A large enterprise (company, hospital etc) has huge numbers of detailed records (sales transaction records, patient records etc), which cannot be stored on-line. Thus, older records are either stored in tertiary storage, or discarded altogether. Saving summary data on-line, and providing a reconstruction algorithm, is an attractive alternative. This sort of technique is at the heart of the proposal in [17]. Managing such data well is a necessary pre-requisite for effective data mining and decision support.

- Statistical databases [19], particularly in conjunction with the DataCube operator [10, 13]: For example, consider Census data with income levels, given as summary tables

[1]The research work described in this paper was motivated by exactly this problem in AT&T. There was interest in estimating daily totals for some data, which historically had been stored aggregated over months. The base data, while available, was several orders of magnitude more voluminous and therefore impractically expensive and time-consuming to handle. If reasonable guesses could quickly be made with respect to the daily totals, these were much preferred. The error could be estimated by computing over the full base data for selected sample aggregates.

(=histograms), with one histogram for each of several attributes (age, years in school, years in present job, geographic location etc). Again, the problem is to recover the detail information, or at least enough of it, so that we can answer combined queries on multiple attributes.

- Scientific databases: For example, consider LANDSAT images with vegetation data over time. Clouds sometimes obscure the view and hide relevant information. The problem is to recover the missing data, exploiting a-priori knowledge (*e.g.*, that vegetation data vary smoothly over space and time).

- Data integration: Two different databases often use different choices of attribute value ranges even for shared attributes. Merging such data requires that values be determined for the intersections of the respective ranges. This information is not directly available in either database and has to be reconstructed. For example, one state may store census data regarding income distribution over ranges 10000-20000, 20000-30000, 30000-40000, and so on. Another state may use a different set of ranges: 15000-25000, 25000-35000, and so on. A company targeting a promotion at some income section of the population may find it convenient to have a single union relation over the two states. Since data has been aggregated over incompatible ranges in the two base relations, such a union cannot easily be created.

In this paper, we show how to attack this reconstruction problem formally. We formulate this as an *inverse problem* (*cf.* [8]) so that we can draw upon the vast array of literature on this topic in the field of signal processing. In particular, we focus on two ways to reconstruct the missing information. One way is to maximize an entropy measure, in which case we show that the estimate is a uniform, piecewise constant function, with the estimates for non-overlapping segments being independent. A second way is to maximize the "smoothness" of the estimate, by means of *linear regularization.* We show experimentally that the linear regularization technique produces excellent results, decreasing the error in the estimate by up to 90% over the naive "uniform" assumption. We further show that these experimental results are not surprising, by means of a Theorem that extends the Shannon-Nyquist Sampling result to our context. In short, we are able to establish formally that for smooth enough sequences, complete recovery of information is possible from a histogram that has computed aggregates over narrow enough ranges.

The paper is organized as follows. In Section 2 we present related work on query optimization and statistical databases. Section 3 gives the mathematical problem formulation. Section 4 presents a brief introduction to the theory of inverse problems and some proposed solutions for database settings. In particular, our central Theorem regarding information recovery is established. In Section 5 we apply the proposed methods on real and realistic (synthetic) data, and report the improvements of our method over naive reconstructions. Section 6 presents extensions of the basic technique to some additional scenarios. Section 7 gives the conclusions and future research directions. In the appendices we present, respectively, some mathematics associated with the proposed linear regularization technique, proof of our central Theorem, as well as the connection between the Shannon-Nyquist (frequency domain) measure of smoothness and the linear regularization notion of smoothness.

2 Survey

There is a lot of related work on query optimization, where the problem is to "guess" the attribute value distribution, to make selectivity estimates for specified queries. Early query optimizers used the uniformity assumption [25], which provably leads to pessimistic results [5]. Modern query optimizers typically use histograms [15]. The histogram of an attribute gives the count of records that fall into each pre-determined sub-range ("bucket") of the attribute range. DeWitt and Muralikrishna [21] examined combined histograms for multiple attributes. Ioannidis and Poosala [15] studied the trade-off between high prediction accuracy and ease of maintenance. Their recommendation was that histograms should maintain perfect information about selected attribute values, and assume the uniform distribution for the rest. A recent, adaptive method, has been suggested by Chen and Roussopoulos [4]. The idea is to approximate the unknown value distribution with a polynomial, and to use query feedback to adjust the coefficients of the polynomial.

Similar approaches have been used for spatial databases: Theodoridis and Sellis [27] suggest a coarse discretization of the address space; for each grid cell, they use the average data density, and, making the uniformity assumption for each individual grid-cell, they estimate the performance of an R-tree.

Related work appeared in statistical databases: Malvestuto [19] examined the case of multiple summary tables, and developed algorithms to determine whether a given query can be evaluated to a single number, a range, or not at all. Ng and Ravishankar [22] also consider multiple summary tables, and propose a matrix-algebra criterion to choose the best combination of summary tables to answer a query.

Incomplete information has been studied extensively. For example, see [14] or [9]. The use of class structure, and other aggregation mechanisms, to store partial information has been presented in [16], and to respond to queries has been studied in [26]. All of these efforts have focussed on the logical nature of partial or missing information. In our paper, there is little qualitative reasoning and the logical analysis is trivial: the emphasis is on effective numerical estimation.

Finally, there is much work on views with aggregates. For instance, [6] and [11] consider how to answer queries using aggregate views, and [12] shows how to maintain such views incrementally. Work along these lines hints at the importance of the problem we consider in this paper, but is not directly relevant to our concerns here.

3 Problem Formulation

The general problem is as follows: Consider a d-dimensional address space, discretized, and consider a function $\mathbf{x}$ on it: $\mathbf{x}[i_1, i_2, \ldots, i_d]$.

The question is: given some partial information about the values of $\mathbf{x}$ and general *a priori* information about the nature of distribution of $\mathbf{x}$ values, what is our best estimate for its value at each point.

Formally, the problem is as follows:

Problem 1 (General under-specified) *Estimate*

$$\mathbf{x}[i_1, i_2, \ldots, i_d] \quad i_j = 1, 2, ... \qquad j = 1, 2, ...d \tag{1}$$

under the constraints

$$C_k(\mathbf{x}) = 0 \quad k = 1, \ldots, n \tag{2}$$

The problem is (typically) under-specified, with n being much smaller than the number of variables. We cannot obtain a unique solution unless we are willing to inject some additional knowledge. This additional knowledge comes in the form of *a priori* information regarding the nature of distribution of $\mathbf{x}$ values, and an error metric for the estimated solution. The problem to be solved then is to minimize this error metric, subject to the given constraints.

Nature of Constraints

The specific constraints can take many different forms, the solutions for most of which are fairly similar.

The simplest constraint is a summation constraint, where we require that the sum of specified $\mathbf{x}$ values be equal to some number. Most "rolled-up" data has this property, for instance, weekly sales totals are obtained as a summation of daily sales totals. Many histograms present counts, which are simple summations, such as the number of times a value within the specific range occurred. For example, the number of employees whose age is between 40 and 44 (inclusive) is the sum of the number of employees aged 40, 41, 42, 43 and 44 respectively.

The other commonly used constraint is an average. Thus, we may have the average temperature recorded for a week, obtained as the average of the average temperatures for each day in the week. Given the total number of $\mathbf{x}$ values being averaged over, converting a summation constraint to an average constraint simply involves a division by a constant.

Averages can sometimes be weighted. We may have average income for a region defined as the average of the average income for the constituent counties, weighted by their respective populations.

When a dimension is projected out, typically a summation (and sometimes an average) is performed on the dimension projected out. Thus, we could have a histogram for the number of employees in each age bracket and a separate histogram for the number of employees in each salary bracket. Each item in either marginal histogram represents a sum of the number of employees with that age (salary) and with each possible salary level (age).

Since all of the constraints described above are fundamentally similar in nature, and most can be transformed from one form to the other in a relatively straightforward manner, we choose to focus on a single well-defined problem for the bulk of this paper.

Also, for simplicity, we concentrate on the 1-d case. Issues with higher dimensions are considered in section 6.1. The matrix $\mathbf{x}$ becomes a vector $\vec{x}$, and the problem becomes:

Problem 2 (1-d under-specified) *Estimate the vector*

$$\vec{x} = [x_i] \quad i = 1, \ldots, N \tag{3}$$

subject to the constraints

$$C_k(\vec{x}) = 0 \quad k = 1, \ldots, n \tag{4}$$

As a point of reference, consider a (time) sequence $\vec{x} = [x_i]$ $i = 1, \ldots, N$ (*e.g.*, count of occurrences of attribute value i or dollars spent on day i). Assume that it is hidden from us; instead, we are given the partial sums (*e.g.*, attribute value histograms or weekly sums) S_k,

Symbol	Definition
N	total number of entries in the vector $\vec{x}$
n	number of constraints
b	batch size
Δx_i	$\equiv x_{i+1} - x_i$: forward difference operator
$\mathcal{F}(\vec{x})$	a functional of the vector $\vec{x}$
$\mathcal{H}(\vec{x})$	entropy function of the given vector ($=-\sum x_i \log x_i$)
$C_k(\vec{x})$	the k-th constraint on the vector $\vec{x}$
$\mathbb{Z}$	the set of signed integers (..., -1, 0, 1, ...)
ω_0	the highest frequency of a signal (in rads per second)

Table 1: Symbols and definitions

$k = 1, \ldots, n$, over contiguous and non-overlapping "batches". To further simplify the notation, in several places we will assume that the sequence is divided into "batches" of equal duration b (*e.g.*, weeks, with $b = 7$).[2]

The available information leads to the following problem formulation:

Problem 3 (partial-sums) *Estimate $\vec{x}$, given*

$$C_k(\vec{x}) \;=\; S_k - \sum_{i=B_{k-1}+1}^{B_k} x_i \;=\; 0 \qquad k = 1, \ldots, n \tag{5}$$

where B_k is the largest value of i included in the k^{th} batch. If all batches are of equal size b, then $B_k \;=\; b * k$.

The question is: given the above information of S_k ($k = 1, \cdots, n$), what is our best estimate for the ("daily") values x_i ($i = 1, \ldots, N$)?

4 Solution Technique

The aim is to minimize a suitable error metric between the estimate and the original vector. While the specific error metric used is not likely to be critical, for the sake of specificity we focus on the root-mean-squared error.

The theory of *inverse problems* [24] is applicable to the question at hand. Our specific case is typically under-constrained and thus ill-posed. Since the original vector is not known, we cannot use the root-mean-squared error as the objective function. We can force a unique solution by requiring minimization (or maximization) of some criterion ("functional") $\mathcal{F}(\vec{x})$, such as the entropy of the vector $\vec{x}$. Then, the problem is well defined:

[2]The up-coming "Linear Regularization" method applies even to *non-contiguous and/or overlapping and/or variable duration batches.* However, contiguous, non-overlapping, and equal duration batches appear most often in practice, and we have chosen to restrict ourselves to this case for the bulk of the paper, both to simplify the mathematical notation and to assist the reader in developing an intuition about the problem.

Problem 4 (General-Regularized) *Estimate $\vec{x}$ to minimize (maximize)*

$$\mathcal{F}(\vec{x})$$

under the constraints

$$C_k(\vec{x}) = 0 \quad k = 1, \ldots, n$$

Under appropriate convexity and continuity conditions, the textbook method for solving both the minimization and the maximization version of such problems is the method of *Lagrange multipliers* [18]. By defining an auxiliary function $\mathcal{L}()$ as

$$\mathcal{L}(\vec{x}; \vec{\lambda}) = \mathcal{F}(\vec{x}) + \sum_{k=1}^{n} \lambda_k * C_k(\vec{x}) \tag{6}$$

In the case of maximization, we have to solve the problem:

$$\max_{\vec{x};\vec{\lambda}} \left(\mathcal{F}(\vec{x}) + \sum_{k=1}^{n} \lambda_k * C_k(\vec{x}) \right) \tag{7}$$

and similarly for minimization, with max replaced by min. We find the required extremum by setting the $(N + n)$ partial derivatives to zero:

$$\frac{\partial}{\partial x_i}(\mathcal{L}(\vec{x}; \vec{\lambda})) = \frac{\partial}{\partial x_i}\mathcal{F}(\vec{x}) + \sum_{k=1}^{n} \lambda_k * \frac{\partial}{\partial x_i} C_k(\vec{x}) \quad i = 1, \ldots, N \tag{8}$$

and

$$\frac{\partial}{\partial \lambda_k}(\mathcal{L}(\vec{x}; \vec{\lambda})) = C_k(\vec{x}) = 0 \quad k = 1, \ldots, n \tag{9}$$

The main question is how to choose the functional $\mathcal{F}()$. The objective is to minimize the expected value of the root-mean-squared error, given what we know a-priori about the distribution of values in the vector.

In the following subsections we describe two popular criteria, namely, *Maximum Entropy* and *Linear Regularization.*

4.1 Maximum Entropy (ME)

Maximum Entropy (*e.g.*, [24, sec. 18.7]) will introduce no additional constraints on the nature of the signal to be estimated. Recall that the entropy of a discrete probability distribution $\vec{p} = [p_1, \ldots, p_n]$ is given by

$$H(\vec{p}) = -\sum_i p_i \log p_i$$

The principle of Maximum Entropy suggests that, for an under-constrained problem, we could make it well-defined by requiring maximization of the entropy. If we know the grand total (sum) of the x_i's, we may assume that the x_i's are non-negative and normalized, so that they add to 1. Then, we have

Problem 5 (partial-sums with ME) *Maximize*

$$\mathcal{F}(\vec{x}) = -\sum_i x_i \log x_i$$

subject to the constraints

$$C_k(\vec{x}) \equiv (S_k - \sum_{i=B_{k-1}+1}^{B_k} x_i) = 0 \qquad k = 1, \ldots, n$$

We can show that the piece-wise constant curve, with $x_p = x_q$ for all p, q in the same "batch", is the solution to problem 5:

Lemma 4.1 *For Problem 5, the Maximum Entropy solution $\vec{x}$ is the piece-wise constant curve.*

Proof: This is an extension of Shannon's classic result that the uniform distribution maximizes entropy *e.g.*, *c.f.* [2]. It can be proven as follows. Consider two values x_p, x_q in the same batch k. Then Eq. 8 gives for p:

$$\begin{aligned} \frac{\partial}{\partial x_p}(\mathcal{L}(\vec{x};\vec{\lambda})) &= 0 \quad \Rightarrow \\ 1 + \log x_p + \lambda_k &= 0 \end{aligned}$$

and similarly for q:

$$\begin{aligned} \frac{\partial}{\partial x_q}(\mathcal{L}(\vec{x};\vec{\lambda})) &= 0 \quad \Rightarrow \\ 1 + \log x_q + \lambda_k &= 0 \end{aligned}$$

exactly because p and q belong to the same batch k, and thus interact only with the constraint C_k.

From the above, we have that

$$\log x_q = -1 - \lambda_k = \log x_p$$

and thus

$$x_p = x_q$$

for every p and q that belong to the same batch (*e.g.*, "week"). **QED**

4.2 Linear Regularization

In many situations, it is expected that there will only be a small difference between successive elements of the vector. Most population distributions, for large enough populations, would follow this principle. Thus, for instance, the distribution of employees across age may follow a "bell-shaped" curve with few very old or very young employees, and a relatively continuous plateau in the middle. We would be surprised if some large company had many 34 year old and 36 year old employees, but very few 35 year old employees, for example.

In such situations, one can require that the solution $\vec{x} = [x_i]$ be smooth by minimizing the functional

$$\mathcal{F}(\vec{x}) = \sum_{i=1}^{N-1} (x_i - x_{i+1})^2 \tag{10}$$

Intuitively, the above functional expresses our belief that the unknown solution $\vec{x}$ is rather smooth; thus, the functional penalizes large squared values for the forward differences $\Delta x_i = x_{i+1} - x_i$. Therefore, the problem becomes: Minimize Eq. 10, subject to the conditions of Eq. 5. The functional of Eq. 10 results in an instance of so-called first order Linear Regularization (or '*Phillips-Twomey method*', or '*constrained linear inversion method*' or '*Tikhonov-Miller regularization*' [24]). In Appendix A we show that this minimization problem leads to a matrix algebra problem, using Lagrange multipliers.

Higher orders of smoothness can be required, by setting the functional $\mathcal{F}()$ to be the sum of squares of higher order forward differences:

$$\Delta^m x_i = \Delta(\Delta^{m-1} x_{i+1} - \Delta^{m-1} x_i) \tag{11}$$

with $\Delta x_i = x_{i+1} - x_i$. Thus, the second order Linear Regularization method has the functional

$$\mathcal{F}(\vec{x}) = \sum_{i=1}^{N-2} (x_i - 2 * x_{i+1} + x_{i+2})^2 \tag{12}$$

For the rest of this work, we focus on first-order Linear Regularization, because (a) it gives as good or better results than higher-order ones (we do not show these experiments, for lack of space) (b) it leads to a simpler matrix problem. Thus, the term "Linear Regularization" will stand for first order Linear Regularization.

4.2.1 Full recovery for smooth signals

A major result in this paper is that we can achieve *full recovery* of information from the summarized data, if the original data is "sufficiently smooth". We achieve this through the use of appropriate spectral filtering, which, as we will demonstrate, can be closely approximated by Linear Regularization. In this subsection we state the Theorem, develop the intuition, and give some arithmetic examples. The formal proof is deferred to Appendix B.

Our Theorem is based on the discrete-time counterpart of the classic Shannon-Nyquist Theorem in signal processing. In plain words, this Theorem is as follows:

> Consider a "slowly varying" discrete-time signal that consists solely of sinusoidal components of period greater or equal to some T_0. This signal can be perfectly reconstructed from sole knowledge of its reduced-rate samples taken once every half-period $T_0/2$ time ticks (or faster).

In the signal processing literature we use $\omega_0 = \frac{2\pi}{T_0}$ to denote the frequency that corresponds to the period T_0. Formally:

Theorem 4.2 (Shannon-Nyquist) *Consider a discrete-time signal $\{x(i)\}_{i\in\mathbb{Z}}$. Assume that its Discrete-Time Fourier Transform (DTFT) $X(e^{j\omega})$ converges, and $X(e^{j\omega}) = 0$, $\frac{\pi}{b} \leq |\omega| \leq \pi$. Then it is possible to reconstruct $\{x(i)\}_{i\in\mathbb{Z}}$ from its reduced-rate samples taken once every b time ticks (or faster).*

Proof: See, e.g., [23]. **QED**

Recall that the DTFT $X(e^{j\omega})$ of a signal $x(i)$ is defined as:

$$X(e^{j\omega}) = \sum_{i=-\infty}^{\infty} x(i)e^{-j\omega i} \tag{13}$$

where $j = \sqrt{-1}$ is the imaginary unit. The DTFT is a periodic function of ω, of period 2π, as is clear from its definition. Thus we focus our attention on the behavior of $X(e^{j\omega})$ over $|\omega| \leq \pi$.

The Shannon-Nyquist Theorem implies that, if we know that a signal $x(i)$ has a smooth distribution over i, then we do not explicitly need to store all the individual values of $x(i)$. It is enough if we store some of them and reconstruct the rest. Common practice is, however, not to keep samples of $x(i)$ but rather histograms of summations or averages of $x(i)$ over specified ranges. Observe that, given an average or sum of $x(i)$ over some range of indices, it is not *a priori* possible to figure out any particular value of $x(i)$ within this range. Thus it appears that our problem is different from the classic Shannon-Nyquist problem. Surprisingly, this is not the case. More specifically, we have the following theorem (stated informally at first):

> Consider a "slowly varying" discrete-time signal that consists solely of sinusoidal components of frequency less than or equal to some ω_0. This signal can be perfectly reconstructed from sole knowledge of its contiguous non-overlapping partial sums taken over π/ω_0 samples at a time (or shorter). Thus, this signal can be fully recovered from an appropriately coarse histogram.

Formally, we have:

Theorem 4.3 *(***Band-limited reconstruction from contiguous non-overlapping partial sums***) Consider a discrete-time signal $\{x(i)\}_{i\in\mathbb{Z}}$. Assume that its Discrete-Time Fourier Transform (DTFT) $X(e^{j\omega})$ converges, and $X(e^{j\omega}) = 0$, $\frac{\pi}{b} \leq |\omega| \leq \pi$. This signal satisfies the condition of the classical Shannon-Nyquist (sub) Sampling Theorem, so it can be recovered from its reduced-rate samples taken every b symbols apart. Then, $\{x(i)\}_{i\in\mathbb{Z}}$ can also be recovered from contiguous non-overlapping partial sums $\{S_k\}_{k\in\mathbb{Z}}$, $S_k = \sum_{i=b(k-1)+1}^{kb} x(i)$, $\forall k \in \mathbb{Z}$*

Proof: See Appendix B. **QED**

For smooth curves, with virtually no high frequency components (with period less than some constant times the width of a single bar in the histogram), Linear Regularization indeed creates an essentially error-free reconstruction. Fig. 1 (a)-(b) shows Linear Regularization and Maximum Entropy, respectively, applied to an approximately Gaussian distribution (more details on this and other datasets are provided in the experiments section; and figures are all grouped together at the end of this paper). The batch size is $b = 8$. This is a very smooth dataset. Linear Regularization provides a visibly better reconstruction than Maximum Entropy. In fact, for the given dataset and batch size, the reconstruction obtained by Linear Regularization is, for all practical purposes, essentially error-free.

A theoretical (and very intuitive) justification of why Linear Regularization performs this well for smooth datasets is obtained by looking at Linear Regularization in the frequency (DTFT) domain. It turns out that Linear Regularization is equivalent to minimizing a spectrally weighted reconstruction energy measure that penalizes high frequency components in

the reconstruction, subject to the summation constraints. Thus, Linear Regularization prefers to allocate reconstructed signal energy in the low frequencies of the spectrum, rather than the high frequencies, and therefore opts for the smoothest (in a particular sense) reconstruction possible under the given summation constraints. This view of Linear Regularization is presented in Appendix C.

5 Experiments

We ran several experiments to evaluate our approach. We used both the Maximum Entropy method and the Linear Regularization method. Since Maximum Entropy is equivalent to the uniformity assumption, we didn't need to do any complicated computations. For the Linear Regularization method the resulting Lagrangian consisted of linear equations; we wrote code to derive the coefficients of these equations and then used an off-the-shelf matrix inversion package (and, specifically, `matlab`) to solve the equations.

The measure of success was the normalized root-mean-square error (RMS), which is a typical measure for forecasting in time series [29]. Specifically, we define:

$$RMS = \left(1/N \sum_{i=1}^{N} (x_i - x_{actual,i})^2 \right)^{1/2} \tag{14}$$

where x_i is the reconstructed value and $x_{actual,i}$ is the actual value at time i.

We ran our experiments on a number of real and synthetic datasets. These are discussed below. Note that two of the datasets are known NOT to be smooth. In an intuitive sense, these are the datasets for which we expect to perform the poorest. What is amazing, as we shall see shortly, is that Linear Regularization still does better than the uniformity (Maximum Entropy) assumption, even for rough datasets.

The four datasets we considered are:

- '**GAUSS**' dataset (synthetic): this dataset has been estimated by drawing samples from a Gaussian distribution and counting the number of samples falling within a given histogram bin. We used N=120 bins. Attribute values, *e.g.*, patient height, patient weight *etc.*, are often distributed as a Gaussian, or some close variant thereof. To the extent that histograms are the typical means of storing attribute value data, this case is typical of the sort of situation in which one can expect the work in this paper to be of value. To make our experiment more realistic, rather than use a perfect Gaussian, we created an "approximate" Gaussian, of the sort one would expect from 20,000 items distributed according to a Gaussian distribution. Thus, the number of values in each bin is a little off from the ideal theoretical value. Furthermore, we normalized the data set to lie between 0 and 1 by diving throughout with the peak value. This is the example used in the previous section; see Figure 1 (Figures are grouped together at the end of this paper).

- '**SINE**' dataset (synthetic): a sinusoid, with N=120 samples: $x_i = \sin(2\pi i/60) \quad i = 0, \ldots, 119$. This is a very smooth dataset.

- '**IBM**' dataset (real): IBM closing prices, from `http://www.ai.mit.edu/stocks.html`. The dataset starts from Aug. 30, 1993, and excludes non-working days. We used the

first 120 values ('**IBM**120' or plain '**IBM**') and the first 240 values ('**IBM**240'). See Figure 3.

- '**LYNX**' dataset (real): Canadian lynx trappings data, 1821-1934, for a total of N=114 samples. This is a well known dataset in population biology - it can be found in any time-sequence book (*e.g.*, [3]), as well as on-line through the "S" statistical package [1]. Notice that it has a periodicity of 9-10 years. However, it is not very smooth: it has abrupt population explosions, with significantly different peak values each time. See Figure 4.

The experiments were designed to answer the following questions:

1. How good is the reconstruction when we use Linear Regularization and Maximum Entropy methods of interpolation, and how does the "smoothness" constraint of Linear Regularization perform against the uniformity assumption (Maximum Entropy), for smooth and "rugged" data?

2. How does the accuracy of reconstruction depend on the length of the "batch"?

3. How does the accuracy of reconstruction depend on the total number of samples N?

These questions are answered next. In the last subsection we also provide guidelines for the practitioner.

5.1 Accuracy

We start by presenting some plots to develop a "feel" for the reconstruction that each method can achieve (recall that figures are grouped together at the end of this paper). The full accuracy comparison results are presented in Tables 2, 3.

Figures 1 (a)-(b) show the reconstruction of the '**GAUSS**' dataset, with Linear Regularization and Maximum Entropy, respectively, for batch size b=8. Figures 2(a)-(b) show the same for the '**SINE**' dataset for batch size b=6. The main point to notice is how well the reconstruction is performed by Linear Regularization. In fact, for the '**GAUSS**' dataset, Linear Regularization plots an almost perfect Gaussian, removing the perturbations in the input dataset. Similarly, for the '**SINE**' dataset, Linear Regularization gives an essentially error-free reconstruction.

Figures 3 and 4 show the reconstructions of the '**IBM**' and the '**LYNX**' datasets, for batch size b=3. Notice that even for these "rugged" datasets, the Linear Regularization method performs well, even visually.

We proceed to a more systematic study of the RMS error of the two competitors. We try several values of the batch size b, and we let each method recover the original dataset.

Table 2 shows the RMS error for the competing methods, for the smooth/synthetic datasets; Table 3 shows the RMS error for the real datasets. In signal processing, it is customary to measure the quality of reconstruction using the "signal-to-noise ratio", that is, the ratio of signal strength (e.g., sample deviation) over RMS error. Therefore, in both Tables, we also report the sample deviation of the respective baseline data.

Linear Regularization consistently outperforms the "uniform" method, as long as the batch size b is consistent with our Theorem 4.3. It is a pleasant surprise that the Linear Regularization

method	Lin. Reg.		Max. Entropy (ME)
dataset	RMS	(rel. savings over ME)	RMS
sinusoid120 b= 2	0.0040463	0.890662	0.037007
sinusoid120 b= 3	0.0080780	0.866208	0.060377
sinusoid120 b= 4	0.0128623	0.844224	0.082569
sinusoid120 b= 5	0.0183893	0.823639	0.104271
sinusoid120 b= 6	0.0246694	0.803691	0.125666
gauss120 b= 2	0.0077611	0.387977	0.0126811
gauss120 b= 3	0.0089781	0.499133	0.0179251
gauss120 b= 4	0.0095857	0.599781	0.0239512
gauss120 b= 5	0.0098519	0.650283	0.0281711
gauss120 b= 6	0.0097175	0.717147	0.0343553
gauss120 b= 8	0.0101855	0.774698	0.0452083

Table 2: RMS errors for each method, and relative savings with respect to the 'uniform=ME'. Batch size b, as specified. The sample deviation of the baseline data is 0.7071 and 0.3485, for the sinusoid and Gaussian data, respectively.

does better even for the '**IBM**' dataset, which is not smooth at all. In fact, being a stock price movement, it is expected to be a "random walk", which is known to be a "fractal", with fractal dimension 1.5 [20]. That is, it is nowhere close to being smooth.

Notice that the savings of the Linear Regularization over the uniformity assumption remain relatively stable for the '**IBM**' dataset with 240 samples.

The relative gains increase with the smoothness of the target sequence, as intuitively expected: the two synthetic, smooth datasets enjoy the best savings (up to 89%), followed by the '**LYNX**' dataset (up to 35% savings - notice that the dataset is somehow periodic), followed by the '**IBM**' dataset, the most 'rugged' of all (savings: up to 21%).

5.2 Dependency on batch size

Figures 5(a)-(b) and 6(a)-(b) plot the RMS error as a function of the batch size b, for the synthetic and real datasets, respectively. Notice that Linear Regularization does consistently better than the uniformity/ME assumption, as long as the batch size b is consistent with our Theorem. The cross-over point for the '**LYNX**' dataset is at $b = 5$, as expected, since the half-period of the major oscillation is 9.5/2=4.25. Notice that for the '**IBM**', '**GAUSS**', and '**SINE**' datasets, Linear Regularization is the consistent winner for a wide range of the b values, because these signals have most of their energy concentrated in low frequencies.

5.3 Dependency on signal length

For both methods, the signal length has small impact on the RMS error. For example, between the '**IBM**' with 120 samples and with 240 samples, the difference in RMS is within 0.04 out of $\approx$0.60, for a range of batch sizes.

method	Lin. Reg.		Max. Entropy (ME)
dataset	RMS	(rel. savings over ME)	RMS
lynx b= 2	387.76	0.35336	599.65
lynx b= 3	676.92	0.26940	926.53
lynx b= 4	927.18	0.19257	1148.31
lynx b= 5	1229.90	0.00750	1239.19
lynx b= 6	1442.58	-0.08837	1325.45
ibm120 b= 2	0.464013	0.084987	0.507111
ibm120 b= 3	0.583036	0.147632	0.684019
ibm120 b= 4	0.664337	0.138739	0.771354
ibm120 b= 5	0.731401	0.181176	0.893233
ibm120 b= 6	0.908743	0.141983	1.05912
ibm120 b= 8	0.891448	0.172178	1.07686
ibm120 b= 10	1.09351	0.217065	1.39668
ibm120 b= 12	1.23193	0.153150	1.45472
ibm240 b= 2	0.446499	0.0593327	0.474662
ibm240 b= 3	0.625589	0.167986	0.751897
ibm240 b= 4	0.698712	0.0314862	0.721427
ibm240 b= 5	0.799962	0.192658	0.990859
ibm240 b= 6	0.928196	0.179626	1.13143

Table 3: RMS errors for each method, and relative savings with respect to the 'uniform=ME'. Batch size b, as specified. The sample deviation of the baseline data is 1578, 5.4999, and 7.6967, for the lynx, ibm120, and ibm240 data, respectively.

5.4 Practitioner's Guide

Linear Regularization results in very small error as long as the batch size b is small enough (smaller than half the period of any significant frequency component). This error is substantially smaller than the error due to a Maximum Entropy reconstruction. As expected, error generally increases with batch size. Once the batch size becomes larger than half the period of a "dominant" frequency, this error starts going up fast. For large batch sizes, there is no clear winner in terms of an interpolation technique.

In light of the above observations, we recommend the use of Linear Regularization if the batch size is less than half the period of any significant frequency component (certainly less than half the period of any dominant frequency component). For larger batch sizes, no recovery technique is a clear winner and so one may as well stick with the uniformity assumption due to its computational simplicity.

6 Extensions

There are several directions in which the work presented above can be extended. We suggest a few such directions below.

6.1 2-d Address Space and OLAP.

The theory of inverse problems can handle 1-d, 2-d and even higher dimensionality address spaces. We have focused mainly on 1-d signals (= time sequences), for two reasons: (a) they lead to a more clear description of the approach and (b) they are very interesting in their own right (sales patterns, stock prices, etc) [3, 29]. However, the reduction in data size becomes particularly emphatic as the number of dimensions is increased, and the techniques presented in this paper become even more important.

Here we show how we could handle higher dimensionalities. Consider the case of a relation with two attributes, such as, *e.g.*, `employees`, with `age` and `salary`. Suppose that we are given one histogram for `age` and another for `salary`, each divided into 3 buckets (ranges). Table 6.1 illustrates the situation

$S_{3,*}$	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$
$S_{2,*}$	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$
$S_{1,*}$	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$
X	$S_{*,1}$	$S_{*,2}$	$S_{*,3}$

Table 4: Illustration of the 2-d case

Thus, we are given the histograms $S_{*,j}$ and $S_{i,*}$ (which correspond to the marginal distributions) and we want to recover the "hidden" values of $x_{i,j}$. The problem is formulated as follows: Given

$$S_{i,*} = \sum_j x_{i,j} \quad i = 1, 2, 3 \tag{15}$$

$$S_{*,j} = \sum_i x_{i,j} \quad j = 1, 2, 3 \tag{16}$$

minimize the functional $\mathcal{F}()$ of choice. Once again, Maximum Entropy corresponds to the independence assumption, that is, if N_{emp} is the total count of employee records

$$x_{i,j} = S_{i,*} * S_{*,j}/N_{emp} \qquad \forall i, j \tag{17}$$

More formally:

Lemma 6.1 *In the 2-d problem above, the Maximum Entropy solution leads to the independence assumption*

Proof: (Sketch). Consider once again Eq. 8, for i, j; let λ_i be the Lagrange multiplier for the constraint of the i-th row, and μ_j the same for the j-th column. Then we have:

$$\mathcal{L}(\vec{x}; \vec{\lambda}) = \mathcal{H}(\vec{x}) + \sum_{i=1}^{3} \lambda_i * (S_{i,*} - \sum_j x_{i,j}) + \sum_{j=1}^{3} \mu_j * (S_{*,j} - \sum_i x_{i,j}) \tag{18}$$

and, setting the partial derivatives to zero:

$$\begin{aligned} \frac{\partial}{\partial x_{i,j}}(\mathcal{L}(\vec{x}; \vec{\lambda})) &= 0 \quad \Rightarrow \\ 1 + \log x_{i,j} + \lambda_i + \mu_j &= 0 \quad \Rightarrow \\ x_{i,j} &= \exp(-1) * \exp(-\lambda_i) * \exp(-\mu_j) \end{aligned} \tag{19}$$

Thus each $x_{i,j}$ is determined as the product of an entity determined purely by the row and an entity determined purely by the column. It is easy to show that the only choices of values for the Lagrangians that will satisfy the marginal distribution constraints are the ones shown in Eq. 17. **QED**

However, if the 2-d joint distribution is smooth, we should do better with Linear Regularization. Specifically, we require that the sum of squares of forward differences (both horizontally and vertically) be minimized:

$$\mathcal{F}(\vec{x}) = \sum_{i,j}(x_{i,j} - x_{i+1,j})^2 + \sum_{i,j}(x_{i,j} - x_{i,j+1})^2 \tag{20}$$

The Lagrange equations will be linear, and the resulting system can be solved exactly, with a matrix inversion package.

The Shannon-Nyquist Sampling Theorem carries over to dimensions higher than one. In this spirit it is also possible to extend our Theorem 4.3 regarding band-limited reconstruction from contiguous non-overlapping partial sums. This observation provides the theoretical justification for extending the Linear Regularization paradigm to higher dimensions, such as in the context of OLAP.

Aside from the database context, 2-d estimation is of use in image processing applications as well. For instance, it is well-known that if an image is "zoomed", say each pixel is replaced by four pixels, that the resulting higher resolution image will be "grainy". In image processing, a local smoothing function is typically used to improve the zoomed image. Linear Regularization can be used to obtain exactly the same effect [7, 28].

6.2 Data Warehousing

It should be clear that the summation constraints $C_k(\vec{x})$ may be arbitrary. Our proposed approach can also handle *overlapping* intervals, as well as intervals of *variable length.* This is especially suitable in the case that we have to merge information from several sources, as in multi-databases and data warehousing [30]. For example, suppose that one source provides weekly (non-overlapping) sums, a second source provides the exact values for some selected days, and a third source provides monthly sums (where month boundaries do not usually coincide with week boundaries). The question is to find the best estimates for the daily values x_i. The problem can be easily formulated:

$$\min_{\vec{x}}(\mathcal{F}(\vec{x})) \tag{21}$$

under the constraints

$$C_{weekly,k}(\vec{x}) = 0 \quad k = 1, ... \tag{22}$$

$$C_{daily,j}(\vec{x}) = 0 \quad j = 1, ... \tag{23}$$

$$C_{monthly,m}(\vec{x}) = 0 \quad m = 1, ... \tag{24}$$

where $\mathcal{F}(\vec{x})$ is a suitable functional, e.g., the Linear Regularization functional.

6.3 Reconstruction of missing values

It should also be stressed that the proposed Linear Regularization approach can handle not only variable length and/or overlapping intervals, but also *missing sums and/or values*, even when the grand total is unknown. Linear Regularization will use the known sums (and/or values), and it will fill-in the missing values, to furnish a smooth curve. Notice that (without a given grand total) Maximum Entropy can not be used *at all* in this case.

7 Conclusions

The main contribution of this work is a formal approach to the recovery of information from summary data, and, more generally, partial data. The idea is to use the machinery of the well-developed "inverse problem theory", to inject a-priori knowledge about the domain, eventually transforming the problem into a constrained optimization problem.
Additional contributions are

- The discovery of the close connection to the classic Shannon-Nyquist sampling theorem, and the formulation and proof of Theorem 4.3: our theorem shows that, for "smooth" enough distributions, it is possible to have full recovery of information, given partial sums.
- The proof (Lemma 4.1) that the Maximum Entropy principle leads to the piecewise constant solution for the one dimensional problem with batch sums available, which perfectly agrees with the uniformity assumption.
- A similar proof (Lemma 6.1) that , for the 2-d problem with marginal sums given, the Maximum Entropy principle leads to the independence assumption.

- Experiments showing that, under the conditions of Theorem 4.3, Linear Regularization consistently outperforms the/uniformity assumption, not only for smooth data, but for "fractal", real data as well (IBM stock price movements, and the lynx trappings dataset).

Future work could examine further ties with the well developed field of inverse problems and image restoration. The interaction between two types of summaries, marginal summations and batching summations, is important for multi-dimensional reconstruction (OLAP), histogram maintenance in query optimization, compression of smooth distributions, and numerous other database applications. We have only scratched the surface here; further work is needed to fully exploit this promising relationship.

Acknowledgments

We thank John Goutsias for his help with the proof of uniformity for the Maximum Entropy solution. We also thank I. S. Mumick for providing feedback on a draft of this paper.

References

[1] Richard A. Becker, John M. Chambers, and Allan R. Wilks. *The New S Language.* Wadsworth & Brooks/Cole Advanced Books & Software, Pacific Grove, CA, 1988.

[2] R.E. Blahut. *Principles and Practice of Information Theory.* Addison-Wesley, Reading, Mass., 1987.

[3] George E.P. Box, Gwilym M. Jenkins, and Gregory C. Reinsel. *Time Series Analysis: Forecasting and Control.* Prentice Hall, Englewood Cliffs, NJ, 1994. 3rd Edition.

[4] Chungmin M. Chen and Nick Roussopoulos. Adaptive selectivity estimation using query feedback. *Proc. of the ACM-SIGMOD*, pages 161–172, May 1994.

[5] S. Christodoulakis. Implication of certain assumptions in data base performance evaluation. *ACM TODS*, June 1984.

[6] Shaul Dar, H.V. Jagadish, Alon Y. Levy, and Divesh Srivastava. Answering SQL queries with aggregation using views. Technical report, AT&T, 1995.

[7] D. E. Dudgeon and R. M. Mersereau. *Multi-Dimensional Digital Signal Processing.* Prentice-Hall, 1984.

[8] Heinz W. Engl, Martin Hanke, and Andreas Neubaue. *Regularization of Inverse Problems.* Kluwer, Dordrecht, 1996.

[9] Georg Gottlob and Roberto Zicari. Closed world database opened through null values. In *Proc. 14th Int'l Conf. on Very Large Databases*, pages 50–61, 1988.

[10] J. Gray, A. Bosworth, A. Layman, and H. Pirahesh. Data cube: a relational aggregation operator generalizing group-by, cross-tab, and sub-totals. Technical Report No. MSR-TR-95-22, Microsoft, 1995.

[11] Ashish Gupta, Venkatesh Harinarayan, and Dallan Quass. Generalized projections: A powerful approach to aggregation. In *Proc. 21st International Conference on VLDB*, Zurich, Switzerland, September 1995.

[12] Ashish Gupta, Inderpal Singh Mumick, and V. S. Subrahmanian. Maintaining views incrementally. In *Proc. of ACM SIGMOD*, Washington, D.C., May 1993.

[13] Venky Harinarayan, Anand Rajaraman, and Jeffrey D. Ullman. Implementing data cubes efficiently. In *Proc. ACM SIGMOD*, pages 205–216, Montreal, Canada, May 1996.

[14] T. Imielinski and W. Lipski. Incomplete information in relational databases. *JACM*, 31(4), October 1984.

[15] Yannis E. Ioannidis and Viswanath Poosala. Balancing histogram optimality and practicality for query result size estimation. *ACM SIGMOD*, pages 233–244, June 1995.

[16] H. V. Jagadish. The incinerate data model. *ACM TODS*, 20(1):71–110, March 1995.

[17] H. V. Jagadish, Inderpal Singh Mumick, and Avi Silberschatz. View maintenance issues in the chronicle data model. In *Proc. ACM PODS*, pages 113–124, 1995.

[18] D. G. Luenberger. *Introduction to Linear and Nonlinear Programming.* Addison-Wesley, Reading, Massachusetts, 1973.

[19] Francesco M. Malvestuto. A universal-scheme approach to statistical databases containing homgeneous summary tables. *ACM TODS*, 18(4):678–708, December 1993.

[20] B. Mandelbrot. *Fractal Geometry of Nature.* W.H. Freeman, New York, 1977.

[21] M. Muralikrishna and David J. DeWitt. Equi-depth histograms for estimating selectivity factors for multi-dimensional queries. *Proc. ACM SIGMOD*, pages 28–36, June 1988.

[22] Wee-Keong Ng and Chinya V. Ravishankar. Information synthesis in statistical databases. *Proc. CIKM*, pages 355–361, November 1995.

[23] Alan Victor Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing.* Prentice-Hall, Englewood Cliffs, N.J., 1989.

[24] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C.* Cambridge University Press, 1992. 2nd Edition.

[25] P.G. Selinger, D.D. Astrahan, R.A. Chamberlain, R.A. Lorie, and T.G. Price. Access path selection in a relational database management system. *Proc. ACM-SIGMOD*, pages 23–34, 1979.

[26] Chung-Dak Shum and Richard Muntz. An information-theoretic study on aggregate responses. *Proc. of VLDB*, pages 479–490, August 1988.

[27] Yannis Theodoridis and Timos Sellis. A model for the prediction of r-tree performance. *Proc. of ACM PODS*, 1996. to appear.

[28] P. P. Vaidyanathan. *Multirate Systems and Filter Banks.* Prentice-Hall, 1993.

[29] Andreas S. Weigend and Neil A. Gerschenfeld. *Time Series Prediction: Forecasting the Future and Understanding the Past.* Addison Wesley, 1994.

[30] Jennifer Widom. Research problems in data warehousing. *CIKM*, November 1995. Invited paper.

A Lagrangian solution to Linear Regularization

The Lagrangian of any of the resulting systems is linear, and can be solved with matrix algebra. For example, for the "partial-sums" problem (problem 3), and for first order Linear Regularization, the method of Lagrange multipliers gives the following matrix equation:

$$\begin{bmatrix} 2 & -2 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & \dots & 0 \\ -2 & 4 & -2 & 0 & \dots & 0 & 0 & 0 & 1 & \dots & 0 \\ 0 & -2 & 4 & -2 & \dots & 0 & 0 & 0 & 1 & \dots & 0 \\ & & & & \ddots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & -2 & 4 & -2 & 0 & \dots & 1 \\ 0 & 0 & 0 & 0 & \dots & 0 & -2 & 2 & 0 & \dots & 1 \\ 1 & 1 & 1 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 & 0 & 0 & \dots & 0 \\ & & & & \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 & 1 & 0 & \dots & 0 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{N-1} \\ x_N \\ \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ S_1 \\ S_2 \\ \vdots \\ S_n \end{bmatrix} \tag{25}$$

B Relationship to the Shannon-Nyquist Theorem in Signal Processing

Here, we present a rigorous proof of Theorem 4.3.

Theorem B.1 *(***Band-limited reconstruction from contiguous non-overlapping partial sum data***) Consider a discrete-time signal* $\{x(i)\}_{i\in\mathbb{Z}}$. *Assume that its Discrete-Time Fourier Transform (DTFT)* $X(e^{j\omega})$ *converges, and* $X(e^{j\omega}) = 0$, $\frac{\pi}{b} \leq |\omega| \leq \pi$. *This signal satisfies the condition of the classical Shannon-Nyquist (sub) Sampling Theorem, so it can be recovered from its reduced-rate samples taken every b symbols apart. Then,* $\{x(i)\}_{i\in\mathbb{Z}}$ *can also be recovered from contiguous non-overlapping partial sums* $\{S_k\}_{k\in\mathbb{Z}}$, $S_k = \sum_{i=b(k-1)+1}^{kb} x(i)$, $\forall k \in \mathbb{Z}$

Proof: The classical Shannon-Nyquist (sub) Sampling Theorem [23] states that if the Discrete-Time Fourier Transform (DTFT) of $\{x(i)\}_{i\in\mathbb{Z}}$ exists (denote it by $X(e^{j\omega})$), and satisfies $X(e^{j\omega}) = 0$, $\frac{\pi}{b} \leq |\omega| \leq \pi$, then it is possible to recover $\{x(i)\}_{i\in\mathbb{Z}}$ from knowledge of its samples, taken every b symbols apart. Here, instead of samples we observe contiguous non-overlapping *partial sums* of a given discrete-time signal, which, in the absence of other information, do not *a priori* permit the reconstruction of any particular sample within the summation interval.

We proceed with the proof by decomposing the summation operation in two steps: one in which "full" partial sum data is obtained; and a second step that involves down-sampling this data to come up with contiguous non-overlapping partial sum data.

Let us define the "full" partial sum data $\{y(i)\}_{i\in\mathbb{Z}}$, $y(i) = \sum_{m=i-b+1}^{i} x(m)$, $\forall i \in \mathbb{Z}$. Observe that $S_k = y(kb)$, $\forall k \in \mathbb{Z}$. We may write $y(i)$ in terms of $x(i)$ as follows:

$$y(i) = \sum_{m=i-b+1}^{i} x(m) = \sum_{m=-\infty}^{\infty} h(i-m)x(m) = (x * h)(i)$$

where $h(i) = 1$, $0 \leq i \leq b-1$, and 0 elsewhere. From the Convolution Theorem for DTFT's, it follows that

$$Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega})$$

where $H(e^{j\omega})$ is the DTFT of $h(i)$, which is easily found to be

$$H(e^{j\omega}) = \frac{1-e^{-j\omega b}}{1-e^{-j\omega}}$$

This is a sinc-like low-pass characteristic that does not vanish except for a total of $b-1$ points $\left\{\frac{2\pi}{b} \times l\right\}_{l=1}^{b-1}$ along the unit circle, and is equal to 1 at the origin. Recall that $X(e^{j\omega}) = 0$, $\frac{\pi}{b} \leq |\omega| \leq \pi$, and $H(e^{j\omega}) > 0$, $|\omega| \leq \frac{\pi}{b}$ (its first zeroes are at $+/-\frac{2\pi}{b}$). Thus, it is possible to reconstruct $X(e^{j\omega})$ (equiv. $\{x(i)\}_{i\in\mathbb{Z}}$) from knowledge of $Y(e^{j\omega})$. This is achieved by filtering $y(i)$ using a filter with characteristic $\frac{1}{H(e^{j\omega})}$, $|\omega| \leq \frac{\pi}{b}$ and 0 elsewhere on the unit circle. Since $H(e^{j\omega})$ does not vanish in this interval, this filter is well defined. Also observe that, by virtue of the above multiplication in frequency, $Y(e^{j\omega})$ obeys the same band-limit as $X(e^{j\omega})$. From the Shannon-Nyquist Theorem [23], it then follows that $\{y(i)\}_{i\in\mathbb{Z}}$ (equiv. $Y(e^{j\omega})$), may be reconstructed based solely on knowledge of its samples, taken b-far apart, i.e., precisely $\{S_k\}_{k\in\mathbb{Z}}$. This is achieved by low-pass filtering a zero-padded version of $\{S_k\}_{k\in\mathbb{Z}}$ using an ideal low-pass filter with cutoff $\frac{\pi}{b}$.

It follows that $\{x(i)\}_{i\in\mathbb{Z}}$ may be recovered from $\{S_k\}_{k\in\mathbb{Z}}$ by filtering a zero-padded version of $\{S_k\}_{k\in\mathbb{Z}}$ using a cascade of an ideal low-pass filter with cutoff $\frac{\pi}{b}$, followed by a filter with characteristic $\frac{1}{H(e^{j\omega})}$, $|\omega| \leq \frac{\pi}{b}$ and 0 elsewhere on the unit circle. This cascade is actually equivalent to the second filter. **QED**

C Frequency domain interpretation of Linear Regularization

An alternative and fruitful view of Linear Regularization is afforded by looking at it in the frequency (DTFT) domain. This Appendix is purely tutorial, and no novelty is claimed here. Recall the Linear Regularization functional:

$$\mathcal{F}(\vec{x}) = \sum_{i=1}^{N-1} (x_i - x_{i+1})^2$$

Switching from vector to (the obvious) sequence notation, we write this as

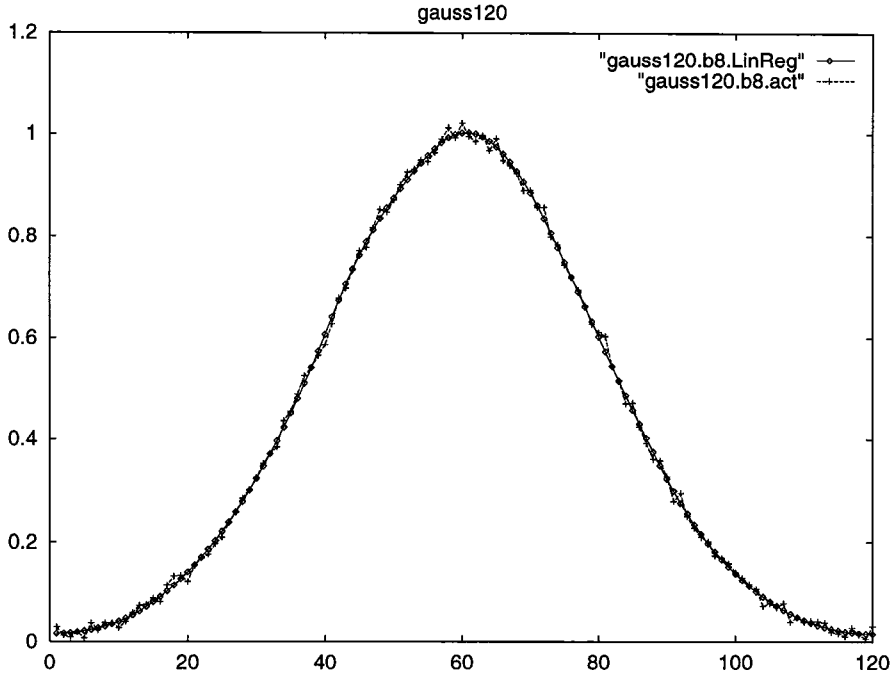
$$\sum_{i=1}^{N-1} (x(i) - x(i+1))^2$$

Let $X(e^{j\omega})$ denote the DTFT of the sequence $x(i)$. From basic DTFT properties, the DTFT of the sequence $x(i+1)$ ($x(i)$ shifted to the left by one) is simply $e^{j\omega}X(e^{j\omega})$ [23]. By linearity,

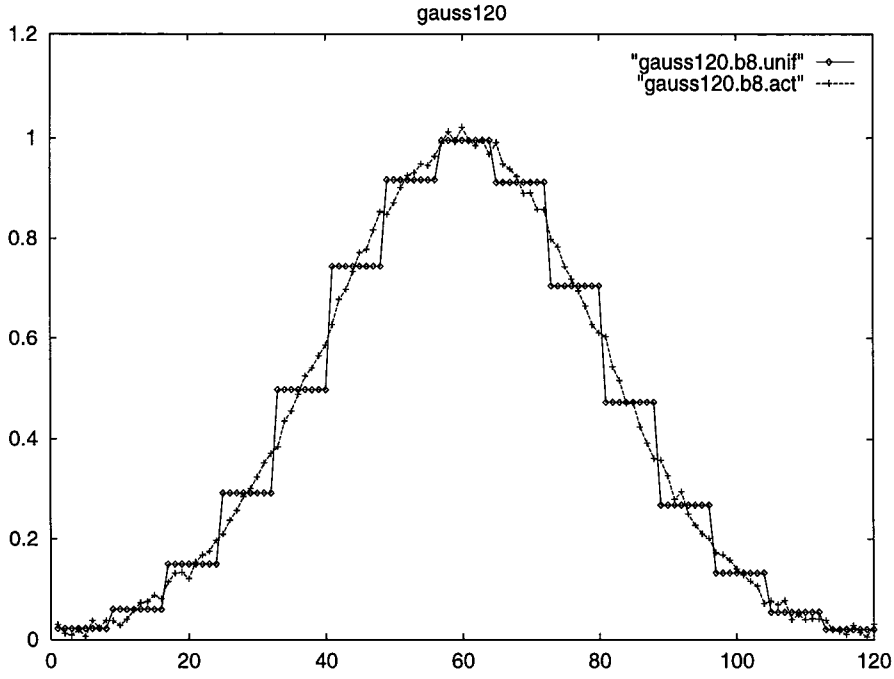
it follows that the DTFT of $x(i) - x(i+1)$ is simply $X(e^{j\omega}) - e^{j\omega}X(e^{j\omega})$, i.e., $(1 - e^{j\omega})X(e^{j\omega})$. From Parseval's Theorem [23], it then follows (modulo negligible edge effects, which can be easily taken care of) that:

$$\begin{aligned}
\sum_{i=1}^{N-1}(x(i) - x(i+1))^2 &= \frac{1}{2\pi}\int_{-\pi}^{\pi}|(1 - e^{j\omega})X(e^{j\omega})|^2 d\omega \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi}|(1 - e^{j\omega})|^2|X(e^{j\omega})|^2 d\omega \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi}|H_{LR}(e^{j\omega})|^2|X(e^{j\omega})|^2 d\omega
\end{aligned}$$

With the obvious definition for $H_{LR}(e^{j\omega})$. Now, $H_{LR}(e^{j\omega})$ can be easily seen to be a high-pass characteristic that is 0 at the frequency origin ($\omega = 0$), small for low frequencies, smoothly approaching the value 2 for high frequencies ($|\omega| \to \pi$), and equal to 2 for $|\omega| = \pi$. This spectral weighting penalizes high frequencies in the reconstruction, and favors low frequencies. Thus Linear Regularization opts for the smoothest (in the above sense) reconstruction consistent with its summation constraints.

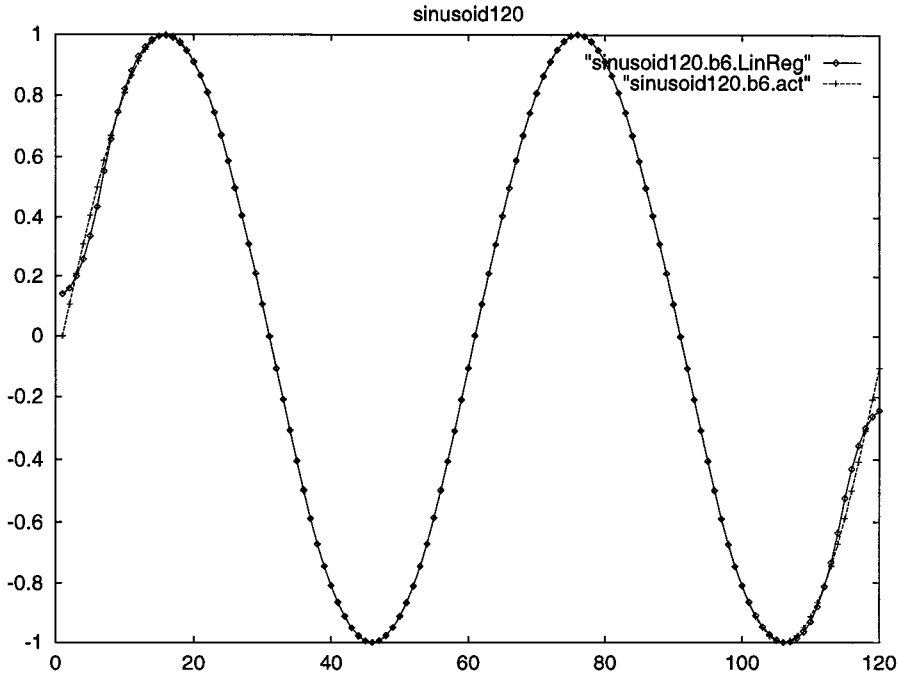


(a) Linear Regularization

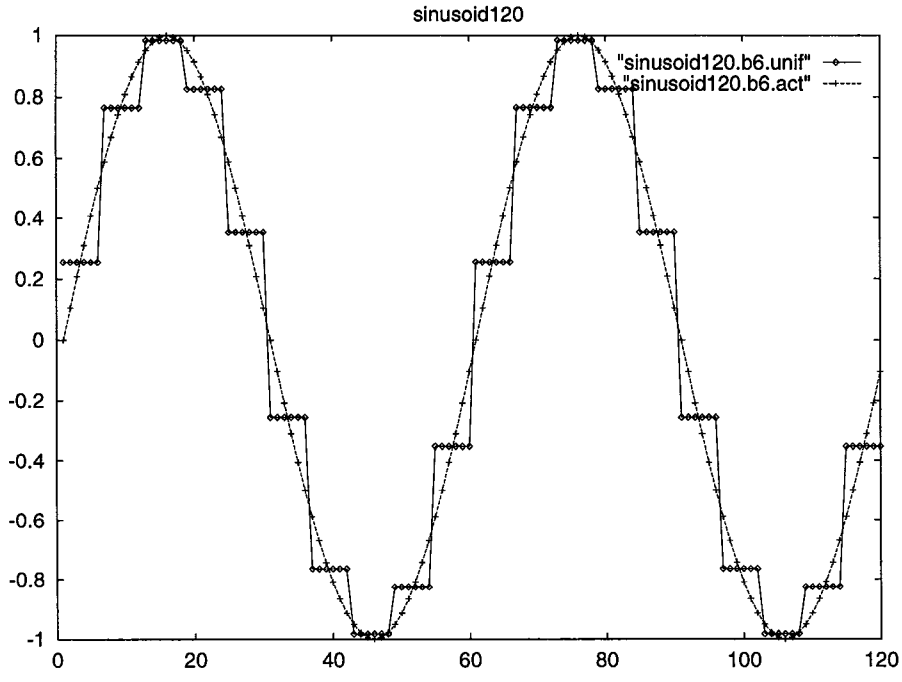


(b) Maximum Entropy

Figure 1: Reconstruction of a Gaussian distribution: with Linear Regularization, we obtain almost error-free reconstruction. Detailed base data: dashed line with "+". Reconstruction: solid line with "diamonds". Batch size b=8.

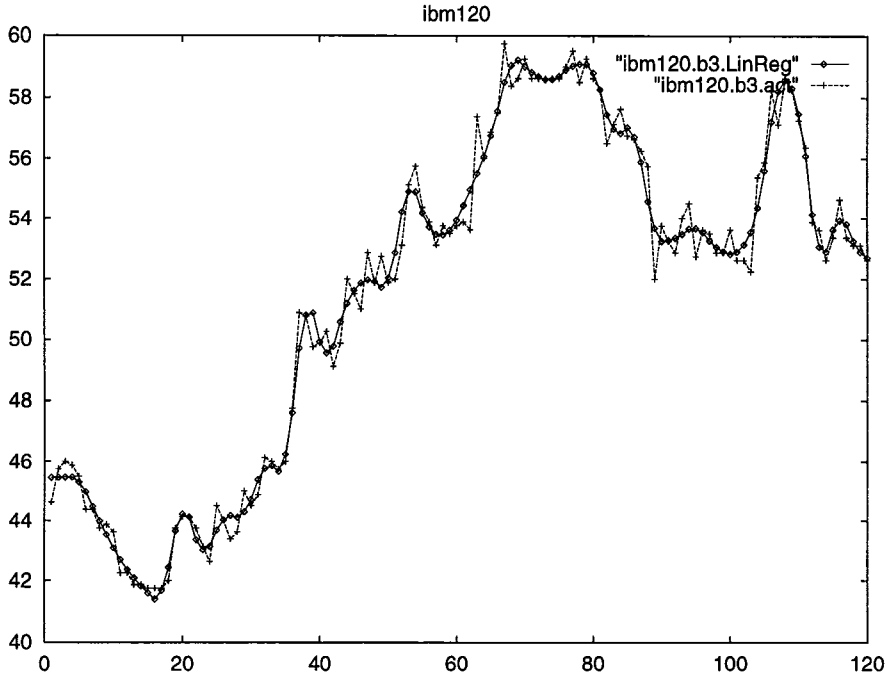


(a) Linear Regularization

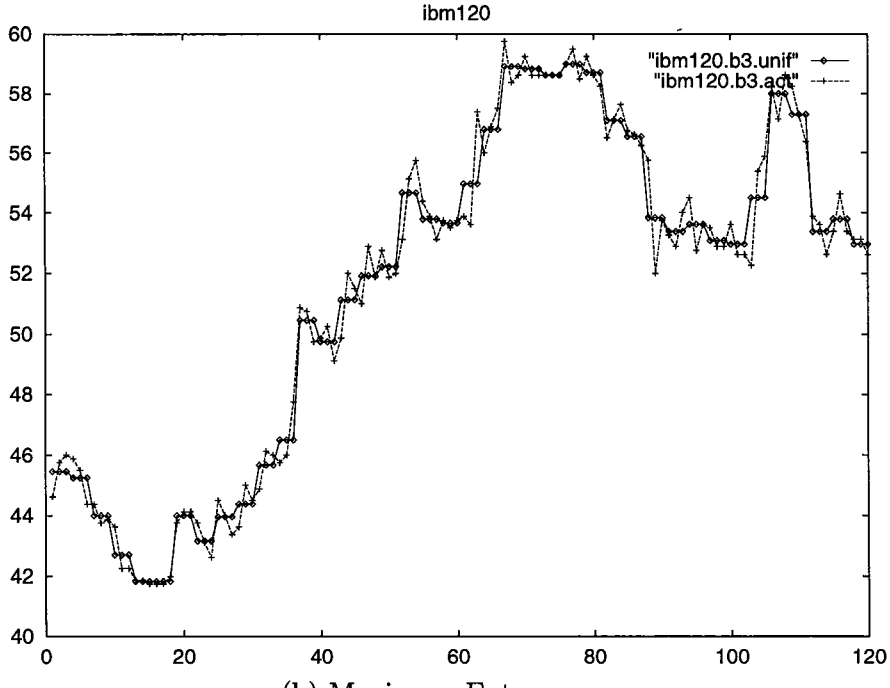


(b) Maximum Entropy

Figure 2: Reconstruction of the '**SINE**' dataset. Detailed base data: dashed line with "+". Reconstruction: solid line with "diamonds". Linear Regularization results in virtually perfect reconstruction, except for the end-effects. Batch size b=6.

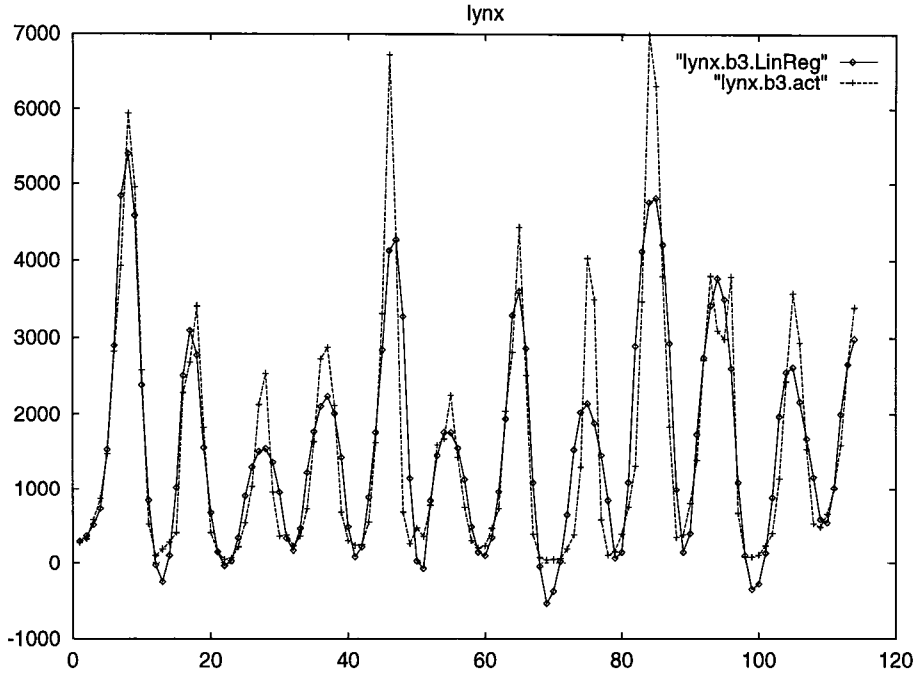


(a) Linear Regularization

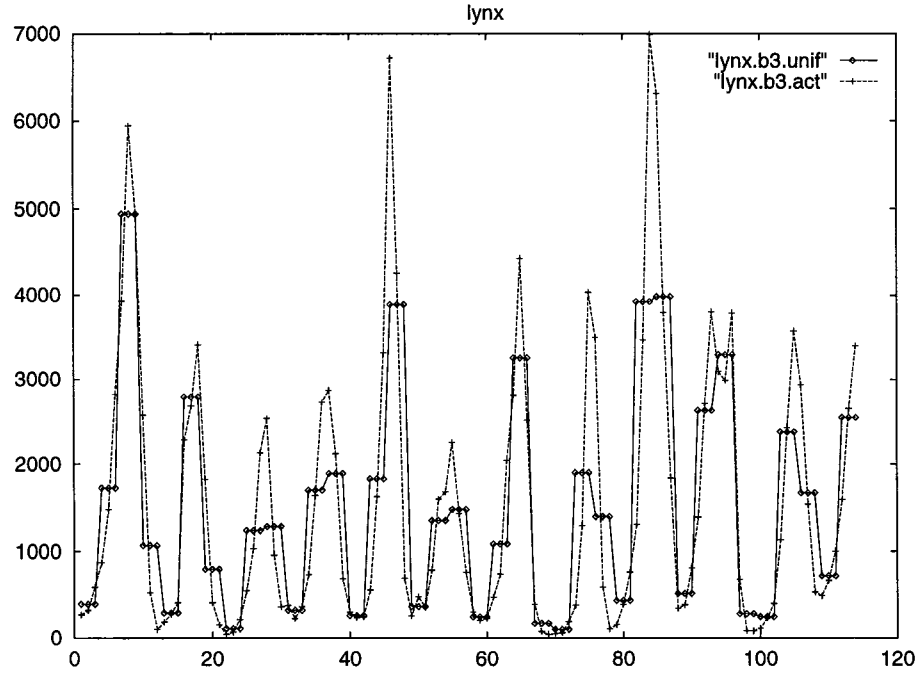


(b) Maximum Entropy

Figure 3: Reconstruction of the '**IBM**' dataset. Detailed base data: dashed line with "+". Reconstruction: solid line with "diamonds". Batch size b=3.

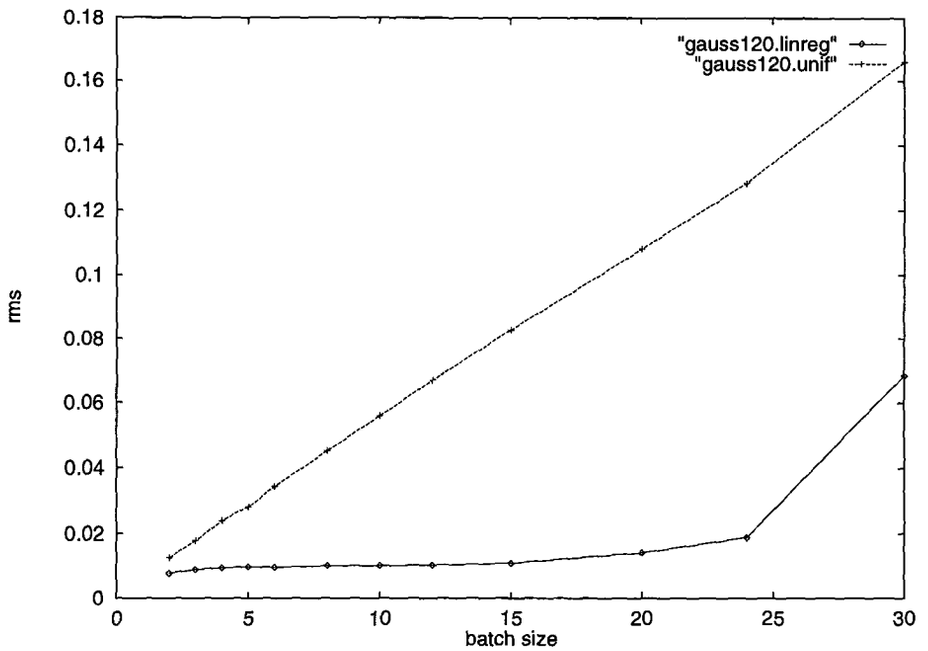


(a) Linear Regularization

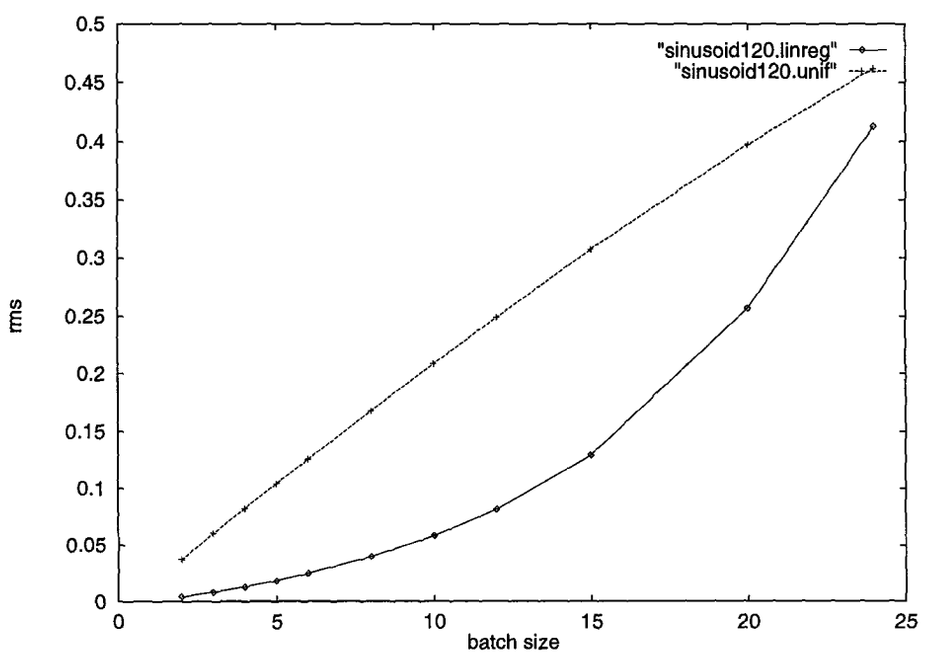


(b) Maximum Entropy

Figure 4: Reconstruction of the '**LYNX**' dataset. Detailed base data: dashed line with "+". Reconstruction: solid line with "diamonds". Batch size b=3.

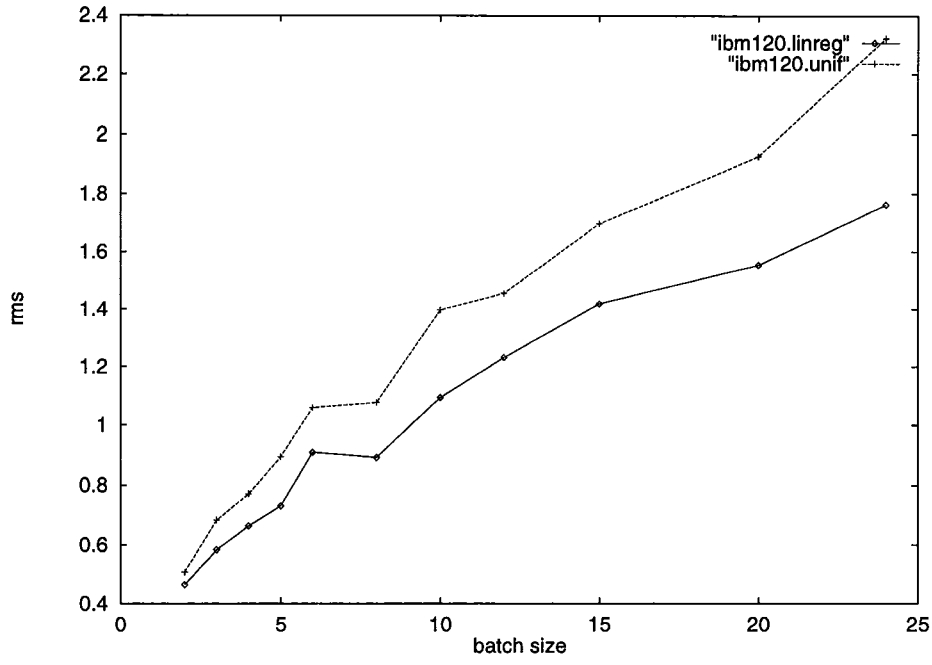


(a) **'GAUSS'**

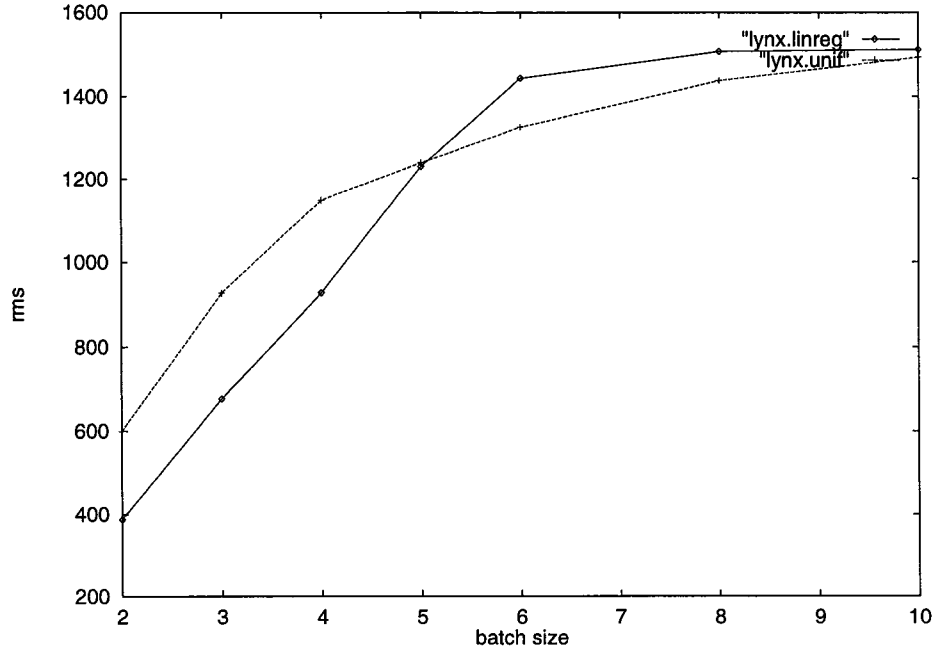


(b) **'SINE'**

Figure 5: RMS error vs batch size b, for the synthetic datasets. Maximum Entropy: dashed line with "+". Linear Regularization: solid line with "diamonds".



(a) **'IBM'**



(b) **'LYNX'**

Figure 6: RMS error vs batch size b, for the real datasets. Maximum Entropy: dashed line with "+". Linear Regularization: solid line with "diamonds".