

Article

An Entropic Approach to Constrained Linear Regression

Argimiro Arratia ^{1,*}  and Henryk Gzyl ^{2,†} ¹ Computer Science, Polytechnical University of Catalonia, 08034 Barcelona, Spain² Centro de Finanzas IESA, Caracas 1010, Venezuela; henryk.gzyl@iesa.edu.ve

* Correspondence: argimiro.arratia@upc.edu; Tel.: +34-934-398-030

† These authors contributed equally to this work.

Abstract: We introduce a novel entropy minimization approach for the solution of constrained linear regression problems. Rather than minimizing the quadratic error, our method minimizes the Fermi–Dirac entropy, with the problem data incorporated as constraints. In addition to providing a solution to the linear regression problem, this approach also estimates the measurement error. The only prior assumption made about the errors is analogous to the assumption made about the unknown regression coefficients: specifically, the size of the interval within which they are expected to lie. We compare the results of our approach with those obtained using the disciplined convex optimization methodology. Furthermore, we address consistency issues and present examples to illustrate the effectiveness of our method.

Keywords: constrained linear regression; Fermi–Dirac entropy; convex optimization; ill-posed inverse problems

MSC: 62J99; 15A29; 62B99

1. Introduction

In the statistical analysis of input–response systems, many prediction or modeling problems involve establishing a functional relationship between input data and output (response) measurements. The input data can be viewed as the control variables of the experiments or as appropriately chosen functions derived from the control data. These inputs are organized into a design matrix X , where each row represents the components of the (transposed) input vector. In this context, the responses to the inputs are treated as real numbers. The linear regression problem consists of solving the following problem.

Find vector $\alpha \in \mathcal{K}_a \subset \mathbb{R}^{K-1}$ and scalar $\beta \in \mathcal{K}_b \subset \mathbb{R}$ such that

$$\begin{aligned} & \text{Solve } \mathbf{y} = \mathbf{X}\alpha + \beta\mathbf{1}_N, \\ & \text{subject to } \alpha \in \mathcal{K}_a = \prod_{j=1}^{K-1} [l_j, L_j], \quad \beta \in \mathcal{K}_b = [h, H] \end{aligned} \quad (1)$$

Here, we temporarily use the standard notations in the statistical literature. Below, we use generic algebraic notations to solve the inverse problem. In this problem, the vector α explains how the different inputs are coupled to obtain the outputs. Thus, the data of the problem consist of the observed outputs, modeled by the vector $\mathbf{y} \in \mathbb{R}^N$ and the $N \times (K - 1)$ design matrix X . The symbol $\mathbf{1}_N$ stands for the N -vector with all components equal to 1 and, as noted, $\beta \in \mathbb{R}$ is a scalar.

The intuitive meaning of the constraint upon α_j amounts to prior knowledge of a range for $\partial\mathbf{y}/\partial X_j$, where X_j is the j -th column of X , i.e., of the sensitivity of the response



Academic Editors: Francisco Chiclana and Paolo Mercorelli

Received: 6 January 2025

Revised: 26 January 2025

Accepted: 28 January 2025

Published: 29 January 2025

Citation: Arratia, A.; Gzyl, H. An Entropic Approach to Constrained Linear Regression. *Mathematics* **2025**, *13*, 456. <https://doi.org/10.3390/math13030456>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

to a change in the input (or control) variable. The specification of \mathcal{K}_b amounts to the specification of a range for the intercept of the regression. These values are either obtained from an underlying model or are to be inferred from the data, as shown in the first example in Section 3.

Let us begin by rewriting the problem as follows. We think of vectors as column vectors, and, for any vector v or matrix A , we shall denote by v^t or A^t , respectively, its transpose; $\langle v, w \rangle = v^t w$ denotes the usual Euclidean scalar product of the vectors v and w ; $\|v\|$ is the Euclidean norm of v ; and $[\cdot \cdot]$ will denote the concatenation of the indicated objects, so, for example, $[A \ v]$ concatenates v as an extra column to matrix A . Let us introduce

$$A_0 = [X \ \mathbf{1}_N], \quad x = [\alpha^t \ \beta]^t \tag{2}$$

Note that A_0 is now a given $N \times K$ matrix. Then, problem (1) can be restated as

$$\begin{aligned} &\text{Solve } y = A_0 x, \\ &\text{subject to } x \in \mathcal{K}_a \times \mathcal{K}_b \subset \mathbb{R}^K. \end{aligned} \tag{3}$$

This is a standard constrained inverse problem. The two standard methods to solve it are as follows. Since the problem may have infinitely many solutions (the kernel of A_0 may be non-0), one way to choose among them is to solve the following variational problem:

$$\begin{aligned} &\text{Find } x^* = \min\{\|x\|^2 : x \in \mathcal{K}_a \times \mathcal{K}_b \subset \mathbb{R}^K\}, \\ &\text{subject to } y = A_0 x \text{ given } y \in \mathbb{R}^N. \end{aligned} \tag{4}$$

Here, $\|x\|$ denotes the standard Euclidean norm in \mathbb{R}^K . A related approach motivated by the fact that the output y may not be in the range of A_0 consists of solving

$$\begin{aligned} &\text{Find } x^* = \min\{d(A_0 x, y) : x \in \mathcal{K}_a \times \mathcal{K}_b \subset \mathbb{R}^K\}, \\ &\text{subject to } y = A_0 x \text{ given } y \in \mathbb{R}^N. \end{aligned} \tag{5}$$

Here, $d(y_1, y_2)$ stands for a distance function on \mathbb{R}^N . Usually, it is the distance derived from the standard Euclidean norm. However, we stress the fact that these may be taken as arbitrary norms. For such approaches in inverse problems and applications, see [1,2], and, for statistics and econometrics, see [3,4].

The consistency condition for problems (3) and (4) is that $y \in \text{int}(\{A_0 x | x \in \mathcal{K}_a \times \mathcal{K}_b\})$ (the interior of the range of the constraint set), which may not be the case due to measurement errors. This leads to the statement of the problem as (5). Instead of this, we propose an extension of (1) or equivalently of (3), which has interesting interpretations. The extended version is stated as

$$\begin{aligned} &\text{Solve } y = X\alpha + \beta\mathbf{1}_N + \epsilon, \\ &\text{subject to } \alpha \in \mathcal{K}_a = \prod_{j=1}^{K-1} [l_j, L_j], \quad \beta \in \mathcal{K}_b = [h, H], \\ &\epsilon \in \mathcal{K}_e = [-d, d]^N. \end{aligned} \tag{6}$$

Two interpretations can be drawn. From the perspective of mathematical programming, ϵ can be viewed as a slack variable that absorbs the discrepancy between the system's response, represented by A_0 , and the observed signal y . Put differently, y now lies within the range of the extended operator A , which acts on the augmented domain $\mathcal{K}_a \times \mathcal{K}_b \times \mathcal{K}_e$.

From a statistical standpoint, Equation (6) can be interpreted as simultaneously determining both the regression coefficients and the measurement noise. This is particularly

relevant because it can help to uncover systematic errors in the measurement process—an important consideration when data are scarce or expensive to obtain. This approach is worth examining alongside the Bayesian framework for linear regression, as described in [5,6], which include numerous applications to classification and machine learning. However, unlike the standard Bayesian approach, which is tied to the methodology in (5) and relies on a parametric (Gaussian) noise model, our proposal is model-free and non-parametric. In contrast, the noise vector estimated through our method can serve as a starting point for the construction of a model of the noise present in the measurements.

To simplify the notations for the mathematical procedure, consider the $N \times (K + N)$ matrix $A = [A_0 \mathbb{I}_N]$, where \mathbb{I}_N stands for the N -dimensional identity matrix, $\xi = [x^t \ \epsilon^t]^t$ for the unknown of the problem, and $\mathcal{K} = \mathcal{K}_a \times \mathcal{K}_b \times \mathcal{K}_e$ for the set of constraints. With this, we restate (6) as the problem

$$\begin{aligned} & \text{Solve } y = A\xi, \\ & \text{subject to } \xi \in \mathcal{K} \end{aligned} \quad (7)$$

Observe that the dimension of the unknown vector is $K + N$, whereas that of the data vector is N , so we have a truly ill-posed problem to solve. Two standard approaches to solving this class of problems were already outlined in (4) and (5). From the abstract point of view, they consist of minimizing a convex function subject to linear and domain constraints, and standard quadratic optimization methods, like *discipline convex optimization* or CVX [7], can be used.

Our proposal is similar in essence but different in detail. We propose to minimize a convex function (the Fermi–Dirac entropy) defined on the constraint set, subject to a linear constraint, that automatically yields an optimizer in the interior of the constraint set. It is in this detail that the difference from quadratic minimization lies.

We also mention that the combination of linear regression and the minimization of the quadratic distance is used in the context of reproducing kernel Hilbert spaces for learning and classification problems, as in the works [8,9]. Besides the standard applications in statistical analysis, and the newer applications in statistical learning already mentioned, we mention the application of linear regression in decoding, as considered by [10].

It is worth noting that our approach provides an alternative to the maximum entropy in the mean methods proposed in [11], as well as its measure-theoretic formulation in [12]. The key difference between our approach and these methods lies in their foundational principles. The maximum entropy in the mean method involves transforming the algebraic problem into one of determining a probability distribution over the set of constraints. In this framework, the solution to the algebraic problem corresponds to the expected value of a random variable with respect to an unknown probability distribution, which is determined using the standard maximum entropy method. In contrast, our approach starts with a Fermi–Dirac-type entropy defined directly on the set of constraints. By minimizing this entropy, we obtain a solution that lies directly within the constraint set.

For optimization purposes, it is useful to note that the entropy function is the Lagrange–Fenchel dual of the logarithm of the Laplace transform of a measure whose support is the constraint space.

The remainder of this paper is organized as follows. In the next section, we collect the mathematical details of the method. Some geometric aspects of the related inverse problem are examined in [13]. At the end of Section 2, we explicitly explain how we measure the quality of the solution. Although the measure is similar to that used in the quadratic optimization procedure, its origin is quite different. The main difference is that our proposal yields a solution (when it exists) in the interior of the constraint set. In Section 3, we consider several variations of the theme of two toy examples: a textbook example and a simulated example. The first of them is an example with a small number of

data points, and we use them to exemplify how to determine the constraint set. We also examine the performance of the method for different sizes of a simulated design matrix and of a data vector, and we compare them to the solutions to the same problems obtained applying the CVX method. In Section 4, we address a consistency issue: we verify that our method is consistent with the obvious algebraic solution when the matrix A is invertible or when it is of full rank with $A^t A$ invertible. We conclude with some remarks.

2. The Entropy Minimization Approach

Clearly, problem (7) is an ill-posed linear inverse problem with convex constraints. Instead of using the traditional least squares methodology, our approach consists of devising a smooth convex function $\Psi(\xi) : \mathcal{K} \rightarrow \mathbb{R}$ and solving

$$\text{Find } \xi^* = \operatorname{argmin}\{\Psi(\xi) : \xi \in \mathcal{K}, \text{ and } A\xi = y\}. \tag{8}$$

To begin with, to avoid excessive notations, we write $\mathcal{K} = \mathcal{K}_a \times \mathcal{K}_b \times \mathcal{K}_e = \prod_{j=1}^M [a_j, b_j]$ with $M = K + N$. The correspondence with the prior notation is clear: the first $K - 1$ labels correspond to \mathcal{K}_a , the K -th to \mathcal{K}_b , and the last N to \mathcal{K}_e . On the Borel sets of \mathcal{K} , we define a measure $dQ(\xi)$ that charges its ‘‘corners’’ as follows:

$$dQ(\xi) = \otimes_{j=1}^M \left(\delta_{a_j}(d\xi_j) + \delta_{b_j}(d\xi_j) \right). \tag{9}$$

We use $\delta_a(dx)$ to denote the unit point mass at the point a (that is, the Dirac delta measure at a). There are three reasons for this proposal. First, the convex hull of the support of $\delta_a(dx) + \delta_b(dx)$ is the interval $[a, b]$. Second, the function $\zeta(\tau) = e^{a\tau} + e^{b\tau}$ introduced in Equation (10) is defined for all $\tau \in \mathbb{R}$. Third, below, we need the invertibility of the mapping $\tau \rightarrow \nabla_\tau M(\tau)$. As the coordinates are separated, this follows from the fact that the equation $\eta = d\zeta(\tau)/d\tau$ has a solution if and only if $\eta \in (a, b)$. Having mentioned these preliminaries, the Laplace transform of dQ is easy to compute:

$$\tau \in \mathbb{R}^M \rightarrow \zeta(\tau) = \int_{\mathcal{K}} e^{\langle \tau, \xi \rangle} dQ(\xi) = \prod_{j=1}^M \left(e^{a_j \tau_j} + e^{b_j \tau_j} \right) \tag{10}$$

Moreover, the moment generating function is defined by

$$M(\tau) = \ln \zeta(\tau) = \sum_{j=1}^M \ln \left(e^{a_j \tau_j} + e^{b_j \tau_j} \right). \tag{11}$$

Then, the function $\Psi(\xi)$ that we seek is defined to be the Lagrange-Fenchel dual of $M(\tau)$, which is given as

$$\Psi(\xi) = \sup_{\tau} (\langle \xi, \tau \rangle - M(\tau)). \tag{12}$$

Making use of the preliminaries mentioned above, a calculation shows that

$$\Psi(\xi) = \sum_{j=1}^M \frac{\xi_j - a_j}{b_j - a_j} \ln \left(\frac{\xi_j - a_j}{b_j - a_j} \right) + \frac{b_j - \xi_j}{b_j - a_j} \ln \left(\frac{b_j - \xi_j}{b_j - a_j} \right) \tag{13}$$

A good reference for these matters is [14]. It is known and standard to verify that $M(\tau)$ is strongly convex and infinitely differentiable in \mathbb{R}^M and $\Psi(\xi)$ is strongly convex and infinitely differentiable in $\operatorname{int}(\mathcal{K})$ and reaches its minimum value in $\operatorname{int}(\mathcal{K})$. Moreover,

for $\boldsymbol{\eta}$ in the interior of \mathcal{K} , the equation $\boldsymbol{\eta} = \nabla \Psi(\boldsymbol{\xi})$ has a unique solution, which is easy to determine analytically. Furthermore, the Lagrange–Fenchel dual of $\Psi(\boldsymbol{\xi})$ is $M(\boldsymbol{\tau})$, and

$$\nabla_{\boldsymbol{\tau}} M(\boldsymbol{\tau}) = (\nabla_{\boldsymbol{\xi}} \Psi)^{-1}(\boldsymbol{\tau}), \quad \text{when } \nabla_{\boldsymbol{\xi}} \Psi(\boldsymbol{\xi}) = \boldsymbol{\tau}. \tag{14}$$

Having made explicit the objects that we need, the solution to problem (8) can be obtained as stated in the following theorem.

Theorem 1. *Let $\Psi(\boldsymbol{\xi})$ be as in (13) and let $M(\boldsymbol{\tau})$ be its Lagrange–Fenchel dual. Suppose that $\boldsymbol{y} \in \text{int}(\mathbf{A}(\mathcal{K}))$. Then, the solution to (8) is given by*

$$\tilde{\boldsymbol{\xi}}_j^* = \frac{a_j e^{a_j(A^t \boldsymbol{\lambda}^*)_j} + b_j e^{b_j(A^t \boldsymbol{\lambda}^*)_j}}{e^{a_j(A^t \boldsymbol{\lambda}^*)_j} + e^{b_j(A^t \boldsymbol{\lambda}^*)_j}}, \quad j = 1, \dots, M. \tag{15}$$

Here, $\boldsymbol{\lambda}^* \in \mathbb{R}^M$ is the point at which $\Sigma(\boldsymbol{\lambda}, \boldsymbol{y}) \equiv \langle \boldsymbol{\lambda}, \boldsymbol{y} \rangle - M(\mathbf{A}^t \boldsymbol{\lambda})$ achieves its maximum value. Moreover,

$$\Psi(\boldsymbol{\xi}^*) = \Sigma(\boldsymbol{\lambda}^*, \boldsymbol{y}). \tag{16}$$

Proof. To obtain the solution (15), we use the standard Lagrange multipliers technique to minimize $\Psi(\boldsymbol{\xi})$, where we first define the Lagrangian function

$$L(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \Psi(\boldsymbol{\xi}) - \langle \boldsymbol{\lambda}, (\mathbf{A}\boldsymbol{\xi} - \boldsymbol{y}) \rangle.$$

Then, equating $\nabla_{\boldsymbol{\xi}} L = 0$ and $\nabla_{\boldsymbol{\lambda}} L = 0$, one obtains the following system for $\boldsymbol{\xi}^*, \boldsymbol{\lambda}^*$:

$$\nabla_{\boldsymbol{\xi}} \Psi(\boldsymbol{\xi}^*) = \mathbf{A}^t \boldsymbol{\lambda}^*, \tag{17}$$

$$\mathbf{A}\boldsymbol{\xi}^* = \boldsymbol{y}. \tag{18}$$

We know that $\nabla_{\boldsymbol{\xi}} \Psi$ is invertible; therefore, from (17) and (14), we obtain

$$\boldsymbol{\xi}^* = (\nabla_{\boldsymbol{\xi}} \Psi)^{-1}(\mathbf{A}^t \boldsymbol{\lambda}^*) = \nabla_{\boldsymbol{\tau}} M(\boldsymbol{\tau}),$$

with $\boldsymbol{\tau} = \mathbf{A}^t \boldsymbol{\lambda}^*$. The equality (16) follows from Corollary 3.3.11 in [14]. \square

Keep in mind that the first K components of $\boldsymbol{\xi}^*$ solve problem (3) and the last N are the estimated measurement error $\boldsymbol{\epsilon}^*$. Moreover, notice that, since $\Sigma(\boldsymbol{\lambda}, \boldsymbol{y}) = \langle \boldsymbol{\lambda}, \boldsymbol{y} \rangle - M(\mathbf{A}^t \boldsymbol{\lambda})$ is a strictly convex, infinitely differentiable function, its maximizer occurs at $\boldsymbol{\lambda}^*$ satisfying

$$\nabla_{\boldsymbol{\lambda}} \Sigma(\boldsymbol{\lambda}, \boldsymbol{y}) = 0 \Leftrightarrow \mathbf{A}\boldsymbol{\xi}^* = \boldsymbol{y}$$

with $\boldsymbol{\xi}^*$ given by (15). In addition, since most numerical software packages are written to solve a minimization problem by default, instead of maximizing $\Sigma(\boldsymbol{\lambda}, \boldsymbol{y})$, it is convenient to minimize $-\Sigma(\boldsymbol{\lambda}, \boldsymbol{y})$. (see <https://metrumresearchgroup.github.io/bbr/> (accessed on 23 January 2025)) for an example. This combines the usual gradient method with a step reduction procedure at each iteration. This is convenient because the objective function may be very flat near the minimum.)

The Reconstruction Error

When one solves problem (8) or its variational version (15) numerically, the solution $\boldsymbol{\xi}^*$ need not satisfy $\mathbf{A}\boldsymbol{\xi}^* = \boldsymbol{y}$ exactly. The reconstruction error simply measures how large the offset is with respect to the problem data. In methods that use $\|\mathbf{A}\boldsymbol{\xi} - \boldsymbol{y}\|^2$, the value of the objective function at the optimum, namely $\|\mathbf{A}\boldsymbol{\xi}^* - \boldsymbol{y}\|^2$, is, at the same time, a measure

of the reconstruction error. In our approach, the minimum value $\Psi(\xi^*)$ does not measure the quality of the reconstruction error. Nevertheless, we know from Theorem 1 that

$$\|\nabla_{\lambda}\Sigma(\lambda, \mathbf{y})\| = \|\mathbf{A}\xi^* - \mathbf{y}\|. \tag{19}$$

So, at the end, the quantitative reconstruction is the same. However, as our method allows us to estimate the additive noise, which is given by the last N components of ξ^* , let us call this vector ϵ^* , and an estimate of the additive measurement error is given by $\|\epsilon^*\| = \|\mathbf{A}_0\mathbf{x}^* - \mathbf{y}\|$.

As we do not make any assumption about the statistical nature of the measurement noise, if it is supposed that the components of ϵ^* are a sample of some random variable, and the number of data points is large, one could use the output of our method combined with the standard statistical methodology to determine the distribution of the noise affecting the measurement process.

3. Numerical Examples

Here, we consider several examples. The first set of examples is built from one taken from the textbook by Stone ([4], Ch. 8), which we examine from several points of view to exemplify the difference between our method and the least squares optimization procedure. We consider various cases depending on whether we assume that there is an experimental error or not and depending on whether the number of data points is larger or smaller than the number of unknown parameters. After these variations, we present a simulated example along the lines suggested in the Introduction for the R package v. 4.4.1 CVXR by [15].

All examples are programmed in R. We use CVXR to implement CVX with the ECOS_BB solver (<https://github.com/embotech/ecos>, accessed on 23 January 2025), which is a branch-and-bound procedure for the solution of mixed-integer convex second-order cone problems. For our optimization method, we use the spectral projected gradient method for large-scale optimization with simple constraints and Barzilai–Borwein step length strategies [16] (R package BB [17]).

3.1. A Textbook Example

To motivate the least squares method, Ref. [4] presents data from a chemical engineering process for polymerization. The data consist of 6 runs of coded temperatures, $h = (-5, -3, -1, 1, 3, 5)$, and the corresponding values of the process, given in the following vector:

$$\mathbf{y}^t = (58.9, 71.6, 78.0, 81.6, 86.0, 85.2) \tag{20}$$

The transpose of the matrix \mathbf{A}_0 is

$$\mathbf{A}_0^t = \begin{pmatrix} -5 & -3 & -1 & 1 & 3 & 5 \\ 25 & 9 & 1 & 1 & 9 & 25 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{21}$$

In fact, the original linear regression problem has a design matrix without the middle row. Since the solution obtained is poor, a non-linear regression of order two is suggested. So, suppose that the underlying model is $y = x_3 + x_1h + x_2h^2$. (We have changed the notations relative to [4] to follow ours.) Our first task is to estimate the ranges for

$$x_3 = y(0), \quad x_1 = \frac{dy}{dh}(0), \quad x_2 = \frac{1}{2} \frac{d^2y}{dh^2}(0).$$

Here, we consider the following possibility: the mid-point for the ranges of x_1 and x_2 is taken to be the average of the numerical computation of these quantities. The consecutive incremental quotients of y are

$$\frac{\Delta y}{\Delta h} = (6.35, 3.2, 1.8, 2.2, -0.4) \Rightarrow \frac{1}{5} \sum \frac{\Delta y}{\Delta h} = 2.63$$

Note that the quotients decrease, suggesting that the curve bends down (that it may be concave), except that there could be a measurement error at the fourth or fifth data point. The average of the incremental quotients of the incremental quotients yields a mid-point for the range of x_2 ,

$$\begin{aligned} \frac{\Delta\left(\frac{\Delta y}{\Delta h}\right)}{\Delta h} &= (-1.57, -0.7, 0.2, -1.3) \\ \Rightarrow \frac{1}{2} \left(\frac{1}{4} \sum \frac{\Delta\left(\frac{\Delta y}{\Delta h}\right)}{\Delta h} \right) &= -0.421. \end{aligned}$$

The mid-point for the range of the intercept is taken to be the average of $y(1)$ and $y(-1)$, which is 79.8. To be conservative, we allow for a 50% miscalculation of the quantities above and obtain that the ranges for x_1 and x_2 are as in Table 1, whereas that for x_3 is obtained by adding (and subtracting) to the mid-point the length of one step with the estimated slope at $h = 0$.

Table 1. Ranges for the unknowns.

x_1	(1.32, 3.94)
x_2	(−0.63, −0.21)
x_3	(77.1, 82.5)

After this preamble, we are ready to consider several possible cases and comparisons.

3.1.1. Data Without Errors

We solve problem (3), $y = A_0x$, with y given as the vector (20), A_0 a 6×3 matrix, whose transpose is shown in (21), and $x = (x_1, x_2, x_3)$ a solution constrained to the box $\mathcal{K} = [1.32, 3.94] \times [-0.63, -0.21] \times [77.1, 82.5]$. Our convex optimization method gives solution $\hat{x} = (2.547, -0.337, 80.815)$, which, not surprisingly, is similar to the one obtained by the least squares method in [4] and also similar to solutions obtained by optimization methods such as CVXR [7,15], which are based on minimizing $\|A_0x - y\|^2$ (see our comments on consistency in Section 4 below). Substituting these estimated coefficients in the second-order polynomial model and evaluating for different h , we can compare our estimated values \hat{y} to the given data values y in Table 2.

Table 2. Estimated values \hat{y} vs. registered values y .

h	−5	−3	−1	1	3	5
y	58.9	71.6	78.0	81.6	86.0	85.2
\hat{y}	59.65	70.14	77.93	83.02	85.42	85.12

3.1.2. Data with Errors

We assume now that the data have measurement errors. We take $d = 20$ to set an ample box for the error (i.e., to be within $[-20, 20]$). So, this time, we use our method to solve problem (7), $y = A_e x$, with y as before and the extended 6×9 matrix $A_e = [A_0 \mathbb{I}_6]$. The solution x is constrained to the same box as in the previous section, and, after solving,

we obtain the estimated values $\hat{x} = (2.611, -0.403, 79.976)$. Inserting these into the polynomial model, we obtain the estimated values \hat{y} for different h . Our method also gives an estimation of the error $\hat{\epsilon}$, and we show all of these quantities, together with the data values, in Table 3.

Table 3. Estimated values \hat{y} and errors vs. registered values y .

h	−5	−3	−1	1	3	5
y	58.9	71.6	78.0	81.6	86.0	85.2
\hat{y}	56.84	68.52	76.96	82.18	84.18	82.95
$\hat{\epsilon}$	2.06	3.08	1.04	−0.58	1.82	2.23

One can see that the estimated \hat{y} plus the estimated error $\hat{\epsilon}$ sum up to the true data value.

Note that the extended problem $y = A_e \xi$ is under-determined and always ill-posed since $N < K + N$ (in our example, $N = 6, K = 3$). Since $A_e = [A_0 \mathbb{I}_N]$, then

$$A_e^t A_e = \begin{pmatrix} A_0^t A_0 & A_0^t \\ A_0 & \mathbb{I}_N \end{pmatrix}$$

The determinant of this block matrix vanishes since

$$\det(A_e^t A_e) = \det(\mathbb{I}_N) \det(A_0^t A_0 - A_0^t \mathbb{I}_N A_0) = 0.$$

Therefore, as mentioned in Section 4, this enhances the convenience of our approach as it does not require inverting matrices or the quest for generalized inverses. We note that, in this case, a method like CVX also succeeds in finding the solution by applying an interior point method, which works well for small and medium-sized problems, but, for a larger problem, it switches to a first-order solver, which can be slow if the problem is not well conditioned [15]. In this particular example, CVXR yields the estimated coefficients $\hat{x}_{cvx} = (2.364, -0.466, 80.974)$; however, the mean square error of the estimated values of y with these coefficients, namely \hat{y}_{cvx} , with respect to the observed values is $mse(\hat{y}_{cvx} - y) = 4.688$, while, for the estimate obtained with our method, we have $mse(\hat{y} - y) = 3.921$. We shall give further proof of the superior precision of our method with respect to CVX in Section 3.2.

3.1.3. Case in Which the Data Are Scarce

We now assume that it is very difficult or costly to measure different values of y and that we only have two values, $y^t = (81.6, 85.2)$, corresponding to $h = 1, 5$, respectively. The transpose of the matrix A_0 is

$$A_0^t = \begin{pmatrix} 1 & 5 \\ 1 & 25 \\ 1 & 1 \end{pmatrix} \tag{22}$$

We keep the bounds for the solution the same as in the previous problems. Note that the problem $y = A_0 x$ is ill-posed since $N < K$ ($N = 2, K = 3$). Then, solving while disregarding errors with our method, we obtain $\hat{x} = (2.967, -0.344, 78.977)$. Solving the same problem with the CVXR method yields the solution $\hat{x}_{cvx} = (3.735, -0.473, 78.337)$. In Table 4, we present the original data values (for all h), the estimations \hat{y} given by the solution with our method, and the estimations \hat{y}_{cvx} obtained with the solution given by CVXR.

Table 4. Estimated values \hat{y} and \hat{y}_{cvx} and registered values y .

h	−5	−3	−1	1	3	5
y	58.9	71.6	78.0	81.6	86.0	85.2
\hat{y}	55.53	66.97	75.66	81.6	84.77	85.2
\hat{y}_{cvx}	47.84	62.87	74.13	81.6	85.29	85.2

Here, our method performs better than CVX when the data are scarce. The values of \hat{x} yield a better estimate of the unobserved values of y . Comparing the mean square errors for both methods and the observed data, we have $mse(\hat{y} - y) = 178.39$ and $mse(\hat{y}_{cvx} - y) = 286.65$.

3.2. A Simulated Example

For this set of examples, we generated Gaussian data with N observations and K predictors, for different values of N and K . Thus, our design matrix A is of size $N \times K$, with entries obtained by generating NK random numbers from a $\mathcal{N}(0, 1)$ random variable. The outcome values are generated by $y = Ax + e$, where $e \in \mathbb{R}^N$ is a sample from a $\mathcal{N}(0, 0.05)$ random variable, and the regression coefficients form an arithmetic progression of difference 1 from $\lceil 1 - K/2 \rceil$ to $\lceil K/2 \rceil$ (e.g., for $K = 10$, $x = (-4, -3, -2, -1, 0, 1, 2, 3, 4, 5)$). The goal is to recover the values of x . We assume that the experimenter has certain knowledge of the range of values for the coefficients and the error, so she sets the constraint boxes as $[-\lceil K/2 \rceil - 1, \lceil K/2 \rceil + 1]$ for the coefficients and $[-1, 1]$ (i.e., parameter $d = 1$) for the error.

We tested this linear system for various values of N and K , considering $N < K$, $N = K$, or $N > K$. Note that, due to the random nature of A and e , for each pair of values of (N, K) , we repeat the reconstruction experiment 100 times and report the average of the square of the norm of the respective errors, namely $\|e\|^2 = \|y - Ax\|^2$ for our method and $\|e_{cvx}\|^2$ for the CVX reconstruction. Moreover, for large values of N and K , it is convenient to scale down the design matrix A and the error e (e.g., multiply by 10^{-2} for the cases of $N \geq 200$). This is to prevent the exploding of the exponentials in the objective function $\Sigma(\lambda, y)$, which could disrupt the convergence of the underlying solver.

Table 5 shows the square of the norms of the reconstructions resulting from the application of our method and that of the CVX method, for various combinations of (N, K) . We see that, in all cases, our method excels CVX in accuracy, with around a 70–90% improvement in most cases.

Table 5. Comparison of the norms of the reconstruction errors for CVX, $\|e_{cvx}\|$, and our method, $\|e\|$, for different values of N and K . The results are averaged over 100 runs.

N	K	$\ e\ ^2$	$\ e_{cvx}\ ^2$
20	10	3.3×10^{-2}	1.8×10^{-1}
20	20	8.6×10^{-2}	5.7×10^{-1}
20	40	9.8×10^{-4}	1.8×10^{-1}
100	100	0.06	1.15
200	100	0.32	1.19
200	300	1.4×10^{-3}	9×10^{-1}
300	200	1.17	10.47
300	300	2.63	11.32
300	500	3.5×10^{-3}	2.3

4. A Consistency Issue

Here, we add a few consistency remarks. A natural question is the following: what does the method yield when A is invertible? Let us apply the standard method of Lagrange multipliers to solve problem (8):

$$\text{Find } \zeta^* = \operatorname{argmin}\{\Psi(\zeta) : \zeta \in \mathcal{K}, \text{ and } A\zeta = \mathbf{y}\}.$$

Form the Lagrangian, $\Psi(\zeta) - \langle \lambda, (A\zeta - \mathbf{y}) \rangle$, and we equate its gradient with respect to ζ to 0 to obtain

$$\nabla_{\zeta}\Psi(\zeta^*) - A^t\lambda^* = 0.$$

We know that $\nabla_{\zeta}\Psi$ is invertible; therefore,

$$\zeta^* = (\nabla_{\zeta}\Psi)^{-1}(A^t\lambda).$$

Consider now the dual problem

$$\text{Find } \lambda^* = \operatorname{argmax}\{\langle \lambda \mathbf{y} \rangle - M(A^t\lambda)\}.$$

The first-order condition for λ^* to be the maximizer is that

$$A\nabla_{\tau}M(A^t\lambda^*) = \mathbf{y} \Leftrightarrow \nabla_{\tau}M(A^t\lambda^*) = A^{-1}\mathbf{y}.$$

Now, invoking (14) and combining the two identities, we obtain that

$$\zeta^* = (\nabla_{\zeta}\Psi)^{-1}(A^t\lambda^*) = A^{-1}\mathbf{y}.$$

Similarly, when A is of full rank and $(A^tA)^{-1}$ exists, a variation of the previous theme yields that $(A^tA)^{-1}A^t\mathbf{y}$ is a solution to $A^tA\zeta = A^t\mathbf{y}$. The matrix $(A^tA)^{-1}A^t$ is a generalized inverse of A .

Clearly, this is not surprising, but it is useful to know that the consistency is maintained in these particular cases. Once more, the strong features of the duality approach, even when A is invertible but ill-conditioned, is very important, as there is no need to invert matrices to numerically find the maximizer of the dual problem.

5. Final Remarks

To summarize, the main features of our approach are as follows: we define a strictly convex function whose domain is the constraint set and whose minimization provides an explicit representation of the solution to problems (3) or its extended version (7). The extended version can be viewed as a regularization of the original problem, designed to address cases where the data do not lie in the range of the operator. Additionally, the approach ensures the tractability of the dual problem for the Lagrange multipliers, which is particularly advantageous from a numerical perspective, as highlighted in the final paragraph of the previous section.

Pending issues to address are as follows. Our approach does not impose any specific assumptions about the measurement errors, besides the fact that they lie within a bounded interval. It would be interesting to explore scenarios where a large number of measurements are feasible, using the output to determine the statistical properties of the measurement error. As demonstrated in Section 3.1.3, the method performs effectively even when data are scarce, but it would be worthwhile to investigate the asymptotic properties of the estimators.

Finally, by employing appropriate vectorization, we can derive a stylized version of the problem addressed in [18] using the method of maximum entropy in the mean. The problem consists of finding an $M \times K$ matrix X and an $N \times K$ matrix E that satisfy

$$Y = A_0X + E. \tag{23}$$

Here, A_0 is a given $N \times M$ matrix, and Y is a given $N \times K$ matrix, and it is also required that the components of X and E satisfy some box constraint. It can be proven that, after vectorization, this problem reduces to a problem equal to (7).

Author Contributions: Conceptualization, methodology, formal analysis, investigation, A.A. and H.G.; software, data curation, A.A.; validation, A.A. and H.G.; writing—original draft preparation, H.G.; writing—review and editing, A.A. and H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The software and data used in this study are available upon request from the first author.

Acknowledgments: We thank the referees for their comments and suggestions that improved the paper. A. Arratia is affiliated with the Soft Computing Research Group (SOCO) at the Intelligent Data Science and Artificial Intelligence Research Center and with the Institute of Mathematics of UPC-BarcelonaTech (IMTech).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bertero, M.; Boccacci, P. *Inverse Problems and Imaging*; Institute of Physics Publishing: Philadelphia, PA, USA, 1998.
2. Engel, H.W.; Hanke, M.; Neubauer, A. *Regularization of Inverse Problems*; Kluwer Academic Publishers: Dordrecht, The Netherlands, 1996.
3. Mittelhammer, R.C.; Judge, G.G.; Miller, D.J. *Econometric Foundations*; Cambridge University Press: Cambridge, UK, 2000.
4. Stone, C.J. *A Course in Probability and Statistics*; Duxbury Press: Belmont, CA, USA, 1996.
5. Rasmussen, C.E.; Williams, C.R. *Gaussian Processes for Machine Learning*; The MIT Press: Cambridge, UK, 2006.
6. Deisenroth, M.P.; Faisal, A.; Ong, C.S. *Mathematics for Machine Learning*; Cambridge University Press: Cambridge, UK, 2020.
7. Grant, M.; Boyd, S.; Ye, Y. Disciplined Convex Programming. In *Global Optimization: From Theory to Implementation*; Springer: New York, NY, USA, 2006; pp. 155–210.
8. Cucker, F.; Zhou, D.-X. *Learning Theory: An Approximation Theory Point of View*; Cambridge University Press: Cambridge, UK, 2006.
9. Paulsen, V.I.; Raghupathi, M. *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces*; Cambridge University Press: Cambridge, UK, 2017.
10. Candes, E.; Tao, T. Decoding by Linear Programming. *IEEE Trans. Inf. Theory* **2005**, *51*, 4203–4215. [[CrossRef](#)]
11. Golan, A.; Judge, G.G.; Miller, D. *Maximum Entropy Econometrics: Robust Estimation with Limited Data*; John Wiley & Sons: New York, NY, USA, 1996.
12. Golan, A.; Gzyl, H. A Generalized Maxentropic Inversion Procedure for Noisy Data. *Appl. Math. Comput.* **2002**, *127*, 249–260. [[CrossRef](#)]
13. Gzyl, H. A Geometry in the Set of Solutions to Ill-Posed Linear Problems with Box Constraints: Applications to Probabilities on Discrete Sets. *J. Appl. Anal.* **2024**. [[CrossRef](#)]
14. Borwein, J.M.; Lewis, A.S. *Convex Analysis and Nonlinear Optimization*, 2nd ed.; CMS-Springer: Berlin/Heidelberg, Germany, 2006.
15. Anqi, F.; Narasimhan, B.; Boyd, S. CVXR: An R-Package for Disciplined Convex Optimization. *J. Stat. Softw.* **2020**, *94*, 1–34.
16. Raydan, M. Barzilai-Borwein Gradient Method for Large-Scale Unconstrained Minimization Problem. *SIAM J. Optim.* **1997**, *7*, 26–33. [[CrossRef](#)]
17. Varadhan, V.; Gilbert, P.D. BB: An R Package for Solving a Large System of Nonlinear Equations and for Optimizing a High-Dimensional Nonlinear Objective Function. *J. Stat. Softw.* **2009**, *32*, 1–26. [[CrossRef](#)]
18. Marsh, T.L.; Mittelhammer, R.; Scott Cardell, N. Generalized Maximum Entropy Analysis of the Linear Simultaneous Equation Model. *Entropy* **2014**, *16*, 825–853. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.