Including Escalation in Cost Estimating Relationships*

Christopher Jarvis†
USAF AFLCMC EGOL/FZC
christopher.jarvis.3@us.af.mil

Abstract

Prior to developing a Cost Estimating Relationship (CER) the data should be normalized for effects not explained by the assumed CER model. This typically includes the effects of inflation and/or escalation which has recently generated considerable discussion within the DoD. In this paper we demonstrate two common CERs models and augment them to include an escalation term. Discussion will be made regarding the feasibility of the solutions of the augmented models.

1 Introduction

When building models in cost estimating, as in many other scientific fields, an analyst begins with observed data and an assumed model form that governs the relationship of the independent and dependent variables. The desire is to determine the value of the parameters associated with the assumed model that provide the best fit in some measure between the model predictions and the observed data. In every application, the observed data needs to be analyzed for applicability and suitableness for usage in deriving the parameters of the model. In the cost estimating environment, the normalization process involves segregating data by type such as recurring or nonrecurring or mapping raw data into consistent data types like hours per unit (HPU) so that appropriate relationships can be investigated. The most common form of data normalization for cost data is to remove the effects of the changing prices due to inflation or escalation. Inflation is defined as the change in the general price level over time [1] while escalation is more narrowly defined to be the change in the price level of a specific good or service over time.

Within recent years, rising costs to develop, procure and maintain weapon systems coupled with increasing Federal budgetary pressures has intensified the scrutiny DoD programs are subjected to during major reviews. This has renewed interest in understanding and accurately accounting for the effects of inflation and escalation so that programs can deliver more credible estimates to Congress, the warfighter and ultimately taxpayers. Through analysis, it has become clear that for many commodities the real price change is increasing faster than inflation. In contrast, some commodities such as electronics, experience a negative yearly real price change. This result is reflected in the 2015 update to Air Force Instruction (AFI) 65-502 [1] which now defines the difference between inflation and specific escalation as well as reaffirm the position that "analysts should use information and methodologies that have the highest probability of accurately estimating the budget authority that will be required".

*Originally presented 9 June 2016 at ICEAA conference, updated with corrections.

†Thank you to the many people who contributed to this work through discussions, comments, helpful edits or by simply enduring my rambling.

Inflation guidance is issued by the Office of the Under Secretary of Defense, Comptroller (OUSD(C)) annually to support the President's Budget request preparations. The inflation guidance contains rates based on the GDP implicit price deflator and is used by each service to build appropriation specific projections of raw and weighted indices [2]. An analyst could independently build a set of rates using a measure other than GDP deflator. Some proposed replacements are the Employment Cost Index (ECI), Consumer Price Index (CPI) and Producer Price Index (PPI). Each of these indices has a focus area and sub-indices which are tailored to specific subsets of those factor areas. The choice of which rate or set of indices to use ultimately depends on the subjectivity of the analyst or local center guidance.

Once a set of inflation or escalation indices is selected and the normalization of data is complete, Cost Estimating Relationships (CERs) are derived to predict future expected costs. Under some conditions, the choice of incorrect inflation or escalation index could result in a lack of fit when developing CERs. Under worse conditions, a choice of an inappropriate inflation or escalation index leads to acceptance of a CER that is misleading and creates a potential for program disconnects.

The point of this paper is to demonstrate how to determine a best fit escalation parameter from existing data by incorporating an escalation term into the CER. This approach has the promise of directly computing an average escalation parameter that does not rely on the subjective choice of the analyst. Furthermore, the solution approach gives insight into the quality of the parameters obtained from the model. In Section 2, an overview of nonlinear regression is presented. In Section 3 and Section 4, examples are presented for two different types of CERs. For each CER, the specific equations will be derived and verification and application of the model performed. Finally in Section 5, a summary of the results and some concluding remarks are made with suggestions for follow-on research.

2 Review of Nonlinear Regression

To facilitate the development of the later equations, a review of nonlinear regression is presented using a matrix approach. For additional introduction and results see [3, 4]. The flexibility of the nonlinear regression approach allows for many different types of error term assumptions and function types.

In terms of notation, $\mathbb{R}$ is the set of all real numbers. Real scalar values and scalar functions are denoted by non-bolded symbols, e.g. $x \in \mathbb{R}$ or $f : \mathbb{R}^n \to \mathbb{R}$. Vectors and vector functions are denoted by bolded symbols, e.g. $\mathbf{x} \in \mathbb{R}^n$ or $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$. Matrices and matrix valued functions are denoted by bold uppercase symbols, e.g. $\mathbf{X} \in \mathbb{R}^{n \times m}$ or $\mathbf{F} : \mathbb{R}^{n \times m} \to \mathbb{R}^{m \times p}$.

Let $\mathbf{x} \in \mathbb{R}^N$ be a data vector, $\boldsymbol{\beta} \in \mathbb{R}^p$ be a vector in the parameter space and $\varepsilon(\mathbf{x}, \boldsymbol{\beta}) \in \mathbb{R}^{N \times p} \to \mathbb{R}$ be the error function associated with the assumed model type. When the function $\varepsilon(\mathbf{x}, \boldsymbol{\beta})$ is linear in $\boldsymbol{\beta}$, the model is called linear. When the function $\varepsilon(\mathbf{x}, \boldsymbol{\beta})$ is nonlinear in $\boldsymbol{\beta}$ or has no linearizing transformation, an approach other than standard Ordinary Least Squares (OLS) must be used to solve for the parameter vector $\boldsymbol{\beta}$. The most common approach of Nonlinear Regression (NLR) involves linearization about a point and iteration. The linearization process creates a linear model that, under the right circumstances, closely approximates the nonlinear model around the specified point. The process is described below.

Assuming that $\varepsilon(\mathbf{x}, \boldsymbol{\beta})$ is sufficiently regular, $\varepsilon(\mathbf{x}, \boldsymbol{\beta})$ can be expanded about the point $\boldsymbol{\beta}^{(k)}$ in the parameter space using a Taylor Series expansion as

$$\varepsilon(\mathbf{x}, \boldsymbol{\beta}) = \varepsilon\left(\mathbf{x}, \boldsymbol{\beta}^{(k)}\right) + \sum_{j=0}^{p-1} \left[\frac{\partial \varepsilon(\mathbf{x}, \boldsymbol{\beta})}{\partial \beta_j}\right]_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(k)}} \left(\beta_j - \beta_j^{(k)}\right) + LTE \tag{1}$$

where LTE is the local truncation error associated with the first order Taylor Series approximation. The term $\beta^{(k)}$ and $\boldsymbol{\beta}^{(k)}$ in this context represents an iterative approximation for the true parameters. Using the Taylor Series approximation above and neglecting the higher order terms, $\varepsilon(\mathbf{x}, \boldsymbol{\beta})$ can be locally approximated as a linear function in $\boldsymbol{\beta}$ and (1) can be written in matrix form as

$$\begin{aligned}\varepsilon(\mathbf{x}, \boldsymbol{\beta}) &\approx \varepsilon\left(\mathbf{x}, \boldsymbol{\beta}^{(k)}\right) + \sum_{j=0}^{p-1}\left[\frac{\partial \varepsilon(\mathbf{x}, \boldsymbol{\beta})}{\partial \beta_j}\right]_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(k)}}\left(\beta_j - \beta_j^{(k)}\right) \\ &= \varepsilon\left(\mathbf{x}, \boldsymbol{\beta}^{(k)}\right) + \left[\frac{\partial \varepsilon(\mathbf{x},\boldsymbol{\beta})}{\partial \beta_0} \quad \frac{\partial \varepsilon(\mathbf{x},\boldsymbol{\beta})}{\partial \beta_1} \quad \cdots \quad \frac{\partial \varepsilon(\mathbf{x},\boldsymbol{\beta})}{\partial \beta_{p-1}}\right]_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(k)}} \begin{bmatrix} \beta_0 - \beta_0^{(k)} \\ \beta_1 - \beta_1^{(k)} \\ \vdots \\ \beta_{p-1} - \beta_{p-1}^{(k)} \end{bmatrix}. \end{aligned} \tag{2}$$

Stacking the equations for each of the data vectors $\mathbf{x}_i$, the matrix system equation given by

$$\mathcal{E}(\mathbf{X}; \boldsymbol{\beta}) = \mathcal{E}\left(\mathbf{X}; \boldsymbol{\beta}^{(k)}\right) + \mathbf{D}^{(k)}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(k)}\right) \tag{3}$$

is obtained, where the model error $\mathcal{E}(\mathbf{X}; \boldsymbol{\beta})$ and the matrix $\mathbf{D}^{(k)}$ of partial derivatives, called the Jacobian are given by

$$\mathcal{E}(\mathbf{X}; \boldsymbol{\beta}) = \begin{bmatrix} \varepsilon(\mathbf{x}_1, \boldsymbol{\beta}) \\ \varepsilon(\mathbf{x}_2, \boldsymbol{\beta}) \\ \vdots \\ \varepsilon(\mathbf{x}_N, \boldsymbol{\beta}) \end{bmatrix} \qquad \mathbf{D}^{(k)} = \begin{bmatrix} \frac{\partial \varepsilon(\mathbf{x}_1,\boldsymbol{\beta})}{\partial \beta_0} & \frac{\partial \varepsilon(\mathbf{x}_1,\boldsymbol{\beta})}{\partial \beta_1} & \cdots & \frac{\partial \varepsilon(\mathbf{x}_1,\boldsymbol{\beta})}{\partial \beta_{p-1}} \\ \frac{\partial \varepsilon(\mathbf{x}_2,\boldsymbol{\beta})}{\partial \beta_0} & \frac{\partial \varepsilon(\mathbf{x}_2,\boldsymbol{\beta})}{\partial \beta_1} & \cdots & \frac{\partial \varepsilon(\mathbf{x}_2,\boldsymbol{\beta})}{\partial \beta_{p-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \varepsilon(\mathbf{x}_N,\boldsymbol{\beta})}{\partial \beta_0} & \frac{\partial \varepsilon(\mathbf{x}_N,\boldsymbol{\beta})}{\partial \beta_1} & \cdots & \frac{\partial \varepsilon(\mathbf{x}_N,\boldsymbol{\beta})}{\partial \beta_{p-1}} \end{bmatrix}_{\boldsymbol{\beta}=\boldsymbol{\beta}^{(k)}}.$$

In the ideal case that an exact solution exists, the term $\mathcal{E}(\mathbf{X}; \boldsymbol{\beta})$ would be zero, hence we want to solve the linear approximate system given by

$$\mathbf{D}^{(k)}\left(\boldsymbol{\beta} - \boldsymbol{\beta}^{(k)}\right) = -\mathcal{E}\left(\mathbf{X}; \boldsymbol{\beta}^{(k)}\right). \tag{4}$$

When the number of data points is not equal to the number of parameters, i.e. $N \neq p$, the $\mathbf{D}$ matrix is non-square and is consequently singular. When there are less data points than parameters ($N < p$) the system (3) does not have a unique solution but rather an infinite number of solutions through a $p - N$ dimensional subspace. When there are more data points than parameters ($N > p$) the system is overdetermined and has either a single solution (in the case of perfect data) or does not have a solution. In contrast, a least squares solution always exists and is equal to the the linear system exact solution if it exists. Note that when there is an equal number of distinct data points and parameters, the system containing the error is solved without redundancy. For this reason, in practical applications, it is typical to have more data points than parameters with the hope that the least squares solution will provide a better approximation to the true population parameters by reducing the influence of the errors.

A mathematical generalization of the matrix inverse is called the pseudoinverse. Given an $[n \times m]$ matrix $\mathbf{A}$, the pseudoinverse of $\mathbf{A}$ is defined by

$$\mathbf{A}^{\dagger} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T$$

where $\mathbf{A}^T$ is the transpose of the matrix. For the overdetermined system,

$$\mathbf{A}\boldsymbol{z} = \boldsymbol{b} \tag{5}$$

the least squares solution using the pseudoinverse is given by

$$\boldsymbol{z} = \mathbf{A}^{\dagger}\boldsymbol{b}. \tag{6}$$

To understand how the pseudoinverse gives the least squares solution note the residual (error) term for any vector solution $\mathbf{z}$ of (5) is given by

$$\mathbf{b} - \mathbf{A}\boldsymbol{z}$$

and the sum of squared errors is

$$(\mathbf{b} - \mathbf{A}\boldsymbol{z})^T (\mathbf{b} - \mathbf{A}\boldsymbol{z}).$$

The least squares solution minimizes the sum of squared errors equation with respect to the solution vector $\boldsymbol{z}$, hence

$$\begin{aligned} 0 &= \frac{\partial}{\partial \boldsymbol{z}} \left[(\mathbf{b} - \mathbf{A}\boldsymbol{z})^T (\mathbf{b} - \mathbf{A}\boldsymbol{z}) \right] \\ &= -2\mathbf{A}^T (\mathbf{b} - \mathbf{A}\boldsymbol{z}). \end{aligned} \tag{7}$$

Rearranging (7) we have

$$\mathbf{A}^T\mathbf{A}\boldsymbol{z} = \mathbf{A}^T\mathbf{b} \tag{8}$$

which is the matrix form of the parameter normal equations. The $\mathbf{A}^T\mathbf{A}$ term on the left side is a square matrix and so long as $\text{rank}(\mathbf{A}) = m$, is invertible. Consequently, the solution is given by

$$\boldsymbol{z} = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{b}$$

which agrees with (6).

Using the pseudoinverse, the system linearized about the point $\boldsymbol{\beta}^{(k)}$ in (3) can be solved for a parameter estimate correction vector. Starting from an initial guess $\boldsymbol{\beta}^{(k)}$ and using (4), we find a new estimate $\boldsymbol{\beta}^{(k+1)}$ by computing

$$\boldsymbol{\beta}^{(k+1)} = \boldsymbol{\beta}^{(k)} + \boldsymbol{\delta}^{(k)} \tag{9}$$

where

$$\boldsymbol{\delta}^{(k)} = -\left[\mathbf{D}^{(k)}\right]^{\dagger} \mathcal{E}\left(\mathbf{X}; \boldsymbol{\beta}^{(k)}\right). \tag{10}$$

This process is known as the Gauss-Newton step and is repeated until the termination criteria are met. A slight modification adds a step scaling parameter $\gamma \in (0, 1]$ to the correction vector, yielding

$$\begin{aligned} \boldsymbol{\beta}^{(k+1)} &= \boldsymbol{\beta}^{(k)} + \gamma\boldsymbol{\delta}^{(k)} \\ &= \boldsymbol{\beta}^{(k)} + \gamma\left(-\left[\mathbf{D}^{(k)}\right]^{\dagger} \mathcal{E}\left(\mathbf{X}; \boldsymbol{\beta}^{(k)}\right)\right). \end{aligned} \tag{11}$$

Since the correction vector is computed from the linearization around the point $\boldsymbol{\beta}^{(k)}$, using a full correction step can overshoot the true solution or produce convergence issues. The scaling parameter allows for partial steps in the right direction and can increase the stability of the overall optimization process, though potentially at the cost of additional iterations required to reach a given tolerance. Typically $\lambda = 0.5$ is sufficient and is used in this paper.

Note that in this derivation, the error term assumption has not been specified as additive, multiplicative or otherwise. This solution process can be applied to any type of model equation so long as the data points are distinct and the model error equation is sufficiently smooth.

3 Radio Example

In this section we analyze a CER for an airborne radio. The assumed form of the CER is given by

$$\text{Cost} = (\beta_0 \cdot \text{Weight} + \beta_1 \cdot \text{Power} + \beta_2 \cdot \text{Frequency} + \beta_3) \cdot (1 + \beta_4)^{\text{year}} \ . \tag{12}$$

The independent variables, Weight, Power, Frequency and Year are all assumed to be significant and the parameter values $\beta_0, \beta_1, \beta_2, \beta_3, \beta_4$ will be determined by the data. When the escalation term is absent and an additive error term is assumed, the model is linear in the parameters and Ordinary Least Squares (OLS) regression could be used to obtain the best fit parameter values. In some cases, a transformation of the data set can allow for OLS to be applied where there is an assumed error term other than an additive error. It has been well documented that these transformations not only introduce bias errors but also distort any confidence regions associated with the parameters [4], [5]. In this paper, a multiplicative error is assumed and nonlinear regression is used to determine the parameter values.

3.1 Parameter solution development

To apply the nonlinear regression solution method introduced in Section 2, the error equation and its partial derivatives need to be defined in terms of the observed data and the assumed model given in (12).

Let y_i be the cost associated with an airborne radio with

$$\mathbf{x}_i = [\text{Weight}_i, \text{Power}_i, \text{Frequency}_i, \text{year}_i]^T$$

characteristics. Given a vector of parameters $\boldsymbol{\beta} = [\beta_0, \beta_1, \beta_2, \beta_3, \beta_4]^T$, the cost CER with multiplicative error is given by

$$\begin{aligned} y_i &= f(\mathbf{x}_i, \boldsymbol{\beta}) \cdot (1 + \varepsilon) \\ &= (\beta_0 \cdot \text{Weight}_i + \beta_1 \cdot \text{Power}_i + \beta_2 \cdot \text{Frequency}_i + \beta_3) \cdot (1 + \beta_4)^{\text{year}} \cdot (1 + \varepsilon) \end{aligned}$$

and the multiplicative error can be written as

$$\varepsilon_i(\mathbf{x}, \boldsymbol{\beta}) = \frac{y_i}{f(\mathbf{x}_i, \boldsymbol{\beta})} - 1. \tag{13}$$

Using the chain rule, the partial derivatives of the error equation are

$$\begin{aligned} \frac{\partial}{\partial \beta_j} \varepsilon(\mathbf{x}, \boldsymbol{\beta}) &= \frac{\partial}{\partial \beta_j} \left[\frac{y_i}{f(\mathbf{x}_i, \boldsymbol{\beta})} - 1 \right] \\ &= \frac{-y_i \left(\frac{\partial}{\partial \beta_j} f(\mathbf{x}_i, \boldsymbol{\beta}) \right)}{f(\mathbf{x}_i, \boldsymbol{\beta})^2}. \end{aligned} \tag{14}$$

From the definition of the CER (12), the partial derivatives are given by

$$\frac{\partial}{\partial \beta_0} f(\mathbf{x}, \boldsymbol{\beta}) = (\text{Weight}_i) \cdot (1 + \beta_4)^{\text{year}_i} \tag{15}$$

$$\frac{\partial}{\partial \beta_1} f(\mathbf{x}, \boldsymbol{\beta}) = (\text{Power}_i) \cdot (1 + \beta_4)^{\text{year}_i} \tag{16}$$

$$\frac{\partial}{\partial \beta_2} f(\mathbf{x}, \boldsymbol{\beta}) = (\text{Frequency}_i) \cdot (1 + \beta_4)^{\text{year}_i} \tag{17}$$

$$\frac{\partial}{\partial \beta_3} f(\mathbf{x}, \boldsymbol{\beta}) = (1 + \beta_4)^{\text{year}_i} \tag{18}$$

$$\frac{\partial}{\partial \beta_4} f(\mathbf{x}, \boldsymbol{\beta}) = (\beta_0 \cdot \text{Weight}_i + \beta_1 \cdot \text{Power}_i + \beta_2 \cdot \text{Frequency}_i + \beta_3) \cdot (1 + \beta_4)^{\text{year}_i - 1} \cdot \text{year}_i. \tag{19}$$

With a parameter vector $\boldsymbol{\beta}^{(k)}$, the partial derivatives of the CER (15)-(19) and (14), a new parameter vector $\boldsymbol{\beta}^{(k+1)}$ can be computed from (11).

3.2 Verification and Test

In this section, the solution method is tested both to verify that the model can identify the correct parameters when there is no error present in the data and to characterize the accuracy when there is error in the data. The independent variable data in this section is notional data based on a DAU example, modified for the purposes of this paper. Perfect data was generated using the parameter values listed in Table 1 and the complete data set is shown in Table 2. Multiplicative errors were randomly sampled from a normal distribution with $\mu = 0$ and $\sigma = 5\%$.

Parameter	Value
β_0	0.060
β_1	0.030
β_2	0.020
β_3	-3.500
β_4	0.018

Table 1: True Radio Dataset Parameters

In Table 3, the list of initial and final values are shown. The final solution was obtained after 75 iterations. Figure 1 shows the parameter values over all iterations and Figure 2 shows the parameter differences between iterations. From these plots, it is evident that after approximately 20 iterations the parameter values are close to the final values. The final computed parameter values are close to the true parameter values, but not exact.

When the true cost data with no error is used, there exists an exact solution to the nonlinear regression problem. In these cases, ideally the computed set of parameters would match the true parameters. In reality, the computed parameters may not match exactly for two reasons. First, when using iterative techniques, stopping criteria are established to determine when enough iterations have been completed. These stopping criteria often include both a maximum number of iterations and some measure of the successive difference between iterations. These settings are within the control of the analyst but require testing to ensure appropriate stopping criteria are used. Second and most important, conditioning of the problem itself combined with finite precision arithmetic means the exact answer may never be able to be computed numerically.

To understand the conditioning of the overall nonlinear regression process, we focus on the sequence of linear system solves in (10). The condition number of a square nonsingular matrix provides a magnification upper bound for the relative error in a computed solution in terms of the

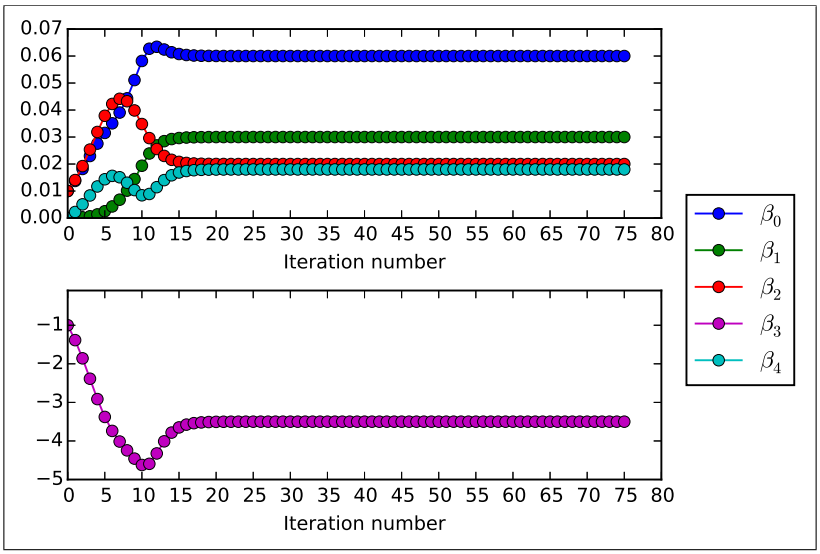

Figure 1: Computed Parameter Values by Iteration

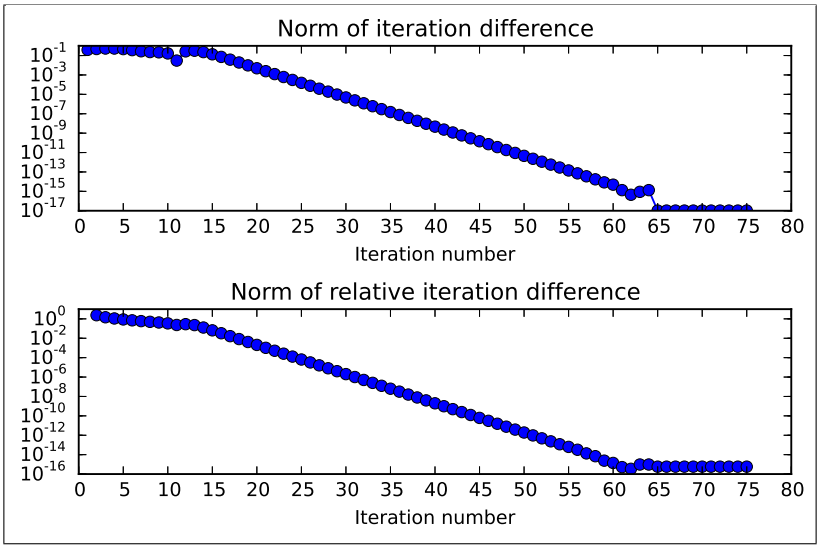

Figure 2: Computed Parameter Difference by Iteration

System	Wgt	Pwr	Freq	Yr	True Cost	Err %	Noisy Cost
1	90	5	76	5	3.90	-8.6%	3.57
2	95	248	148	6	14.02	1.2%	14.19
3	65	73	41	4	3.66	2.8%	3.76
4	118	177	255	10	16.72	-0.3%	16.68
5	85	285	37	9	12.79	7.2%	13.70
6	36	116	268	1	7.64	7.8%	8.23
7	93	195	188	2	12.11	0.7%	12.20
8	134	8	280	4	11.15	-9.0%	10.15
9	95	73	380	7	13.58	-5.6%	12.82
10	146	373	344	8	26.91	7.0%	28.78
11	76	324	303	4	18.09	3.9%	18.78
12	32	232	155	1	8.63	1.9%	8.80
13	34	151	190	2	7.12	-5.3%	6.74
14	146	247	321	8	22.02	1.6%	22.36
15	50	47	277	1	6.57	-4.4%	6.28

Table 2: Radio Example Dataset

Parameter	Initial Value	Final Value	Final Error	Relative Errors
β_0	0.010	0.060	1.92e-11	3.20e-10
β_1	0.000	0.030	-1.93e-12	-6.43e-11
β_2	0.010	0.020	4.53e-12	2.27e-10
β_3	-1.000	-3.500	-1.26e-09	3.61e-10
β_4	0.000	0.018	-1.56e-11	-8.69e-10

Table 3: Computed Parameters From Radio Dataset Without Error

relative error in the data [6]. As an illustration, suppose that $\mathbf{A}$ is non-singular matrix, $\mathbf{z}$ satisfies $\mathbf{Az} = \mathbf{b}$, $\tilde{\mathbf{z}}$ satisfies the perturbed data problem $\tilde{\mathbf{A}}\tilde{\mathbf{z}} = \tilde{\mathbf{b}}$ and $\|\tilde{\mathbf{A}} - \mathbf{A}\|\|\mathbf{A}^{-1}\| < 1$, then

$$\frac{\|\mathbf{z} - \tilde{\mathbf{z}}\|}{\|\mathbf{z}\|} \leq \frac{\kappa(\mathbf{A})}{\left(1 - \|\tilde{\mathbf{A}} - \mathbf{A}\|\|\mathbf{A}^{-1}\|\right)} \left(\frac{\|\mathbf{b} - \tilde{\mathbf{b}}\|}{\|\mathbf{b}\|} + \frac{\|\mathbf{A} - \tilde{\mathbf{A}}\|}{\|\mathbf{A}\|} \right) \tag{20}$$

where $\kappa(\mathbf{A}) = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|$ is defined to be the condition number of the matrix $\mathbf{A}$. In practical applications, only perturbed data problems are actually solved due to the errors in the collected data or conversion to finite precision arithmetic. Even if the collected data were perfect, i.e. $\|\mathbf{b} - \tilde{\mathbf{b}}\| = 0$, perturbation of the system given by $\|\mathbf{A} - \tilde{\mathbf{A}}\|$ could still produce an error. The condition number for perfectly conditioned systems is 1, hence the relative error in the computed solution is no greater than the error in the data. As the condition number of a matrix grows, the error in the computed solution may increase proportionally. A general rule of thumb says that for $\kappa(\mathbf{A}) = 10^k$, a loss of k digits of accuracy is possible. For matrices that are singular, $\mathbf{A}^{-1}$ does not exist and no exact solution to the system exists, thus the condition number is taken to be $\kappa(\mathbf{A}) = \infty$. The matrix condition number concept can be extended to matrices that are either singular or non-square. In these cases, the pseudoinverse is used to solve the system in a least squares sense and the generalized condition number is given as $\kappa(\mathbf{A}) = \|\mathbf{A}\| \cdot \|\mathbf{A}^{\dagger}\|$ in [7] or equivalently as

$$\kappa(\mathbf{A}) = \frac{\sigma_1}{\sigma_r}$$

where σ_1 and σ_r are the largest and smallest non-zero singular values of $\mathbf{A}$.

The condition number of the regression matrices are useful for detecting and understanding any multicollinearity that may be present in the dependent variable datasets and model specification. In Figure 3, we plot the condition number of the error function Jacobian for each iteration. The condition number of the final iteration Jacobian is around 1,200, however the initial iteration condition number is over 30,000 indicating that the initial parameter vector guess produced an ill-conditioned system. Ultimately, the method does produce sufficiently close estimates of the parameters.

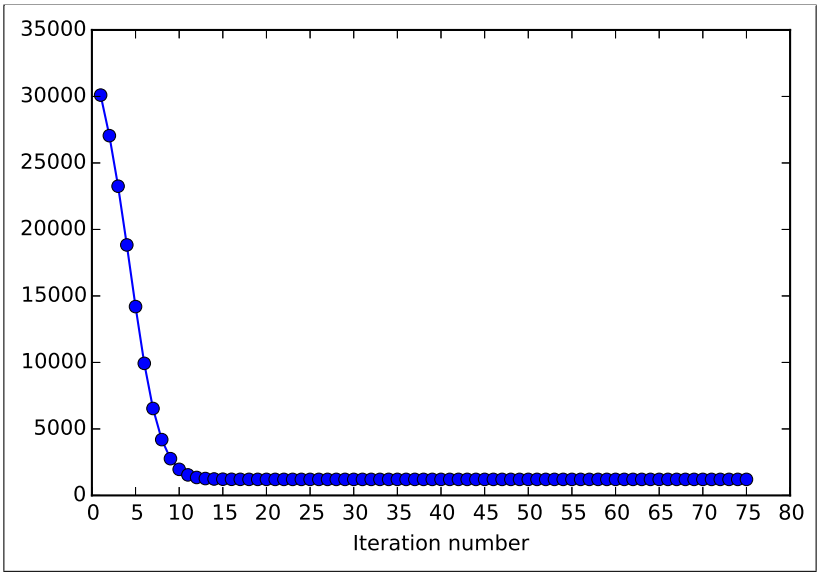

Figure 3: Jacobian Condition Number by Iteration

In Table 2, the Noisy Cost data includes an error term randomly sampled from a Normal distribution. In Table 4, the initial guess and final computed parameter values are shown. Figure 4 shows the parameter values over all iterations and Figure 5 shows the parameter differences between iterations. Again, after approximately 20 iterations the computed parameter values do not change much.

Parameter	Initial Value	Final Value	Final Error	Relative Errors
β_0	0.010	0.055	5.49e-03	9.14e-02
β_1	0.000	0.032	-2.39e-03	7.96e-02
β_2	0.010	0.019	1.30e-03	6.51e-02
β_3	-1.000	-3.280	-2.20e-01	6.29e-02
β_4	0.000	0.022	-3.63e-03	2.01e-01

Table 4: Computed Parameters From Radio Dataset With Error

In Figure 6, the condition number of the error function Jacobian for each iteration is plotted. The condition number of the Jacobian matrix iterates is similar to the perfect data case. Interestingly, the erroneous cost data leads to a set of computed parameters that actually lower the condition number of the Jacobian evaluations. Due to the errors in the cost data, the final parameters have a non-negligible difference from the true parameters. Specifically, the escalation parameter has the largest overall error, about 20% compared to the true parameter value.

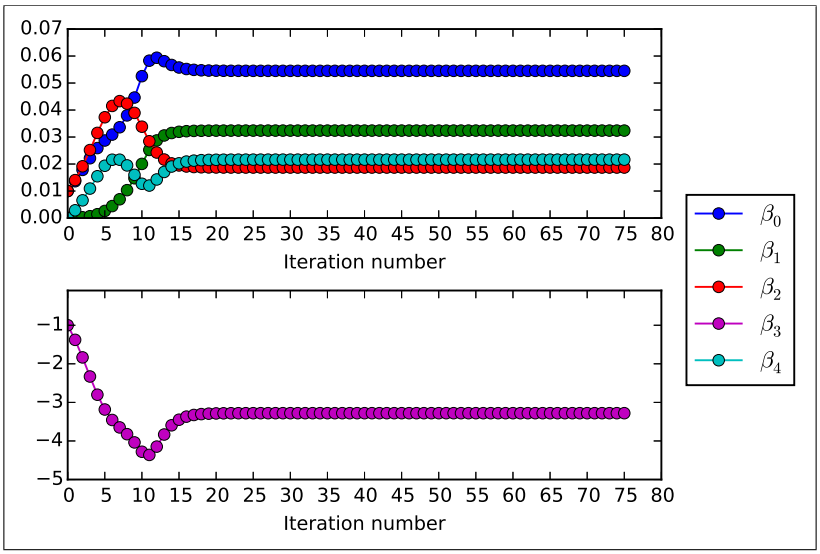

Figure 4: Computed Parameter Values by Iteration

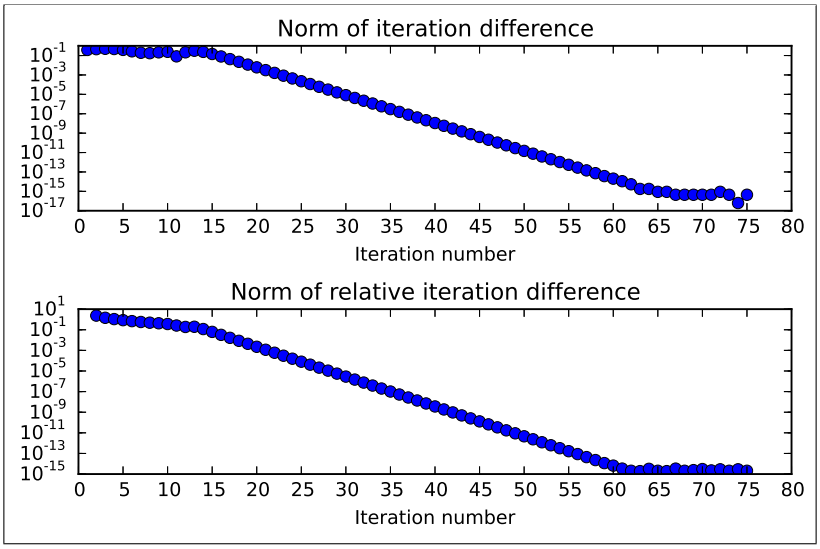

Figure 5: Computed Parameter Difference by Iteration

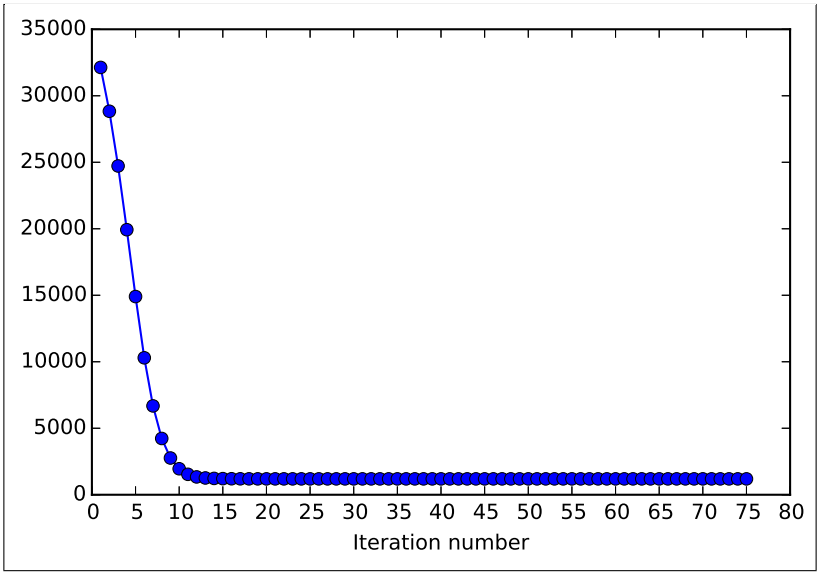

Figure 6: Jacobian Condition Number by Iteration

In Table 5, the degree of linear dependence of the parameters can be seen by examining the parameter correlation matrix. The parameters $\beta_0, \beta_1, \beta_2$ for this problem are coefficients for the Weight, Power and Frequency characteristics respectively and β_3 is the intercept coefficient for the linear part of the model. The escalation parameter does exhibit a strong correlation with some of the other parameters, but not unreasonably high. It is worth noting here that eigenvalues of the parameter correlation matrix are related to the singular values that determine the condition number of the linear system. Therefore, if the parameters are very highly linearly correlated, the effect can be observed in a very high condition number of the system.

	β_0	β_1	β_2	β_3	β_4
β_0	1.000				
β_1	0.757	1.000			
β_2	0.259	0.102	1.000		
β_3	-0.956	-0.777	-0.421	1.000	
β_4	-0.856	-0.821	-0.316	0.805	1.000

Table 5: Parameter Correlation Matrix for Radio Dataset With Error

4 Cost Improvement Curve Example

In this section, a CER for a cost improvement curve problem is analyzed. The Average Unit Cost (AUC) for a production lot of missiles using unit theory learning with rate is given by

$$\text{AUC} = \beta_0 \left(\frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} - \left(F - \frac{1}{2}\right)^{\beta_1+1}}{\beta_1 + 1} \right) \cdot (L - (F-1))^{\beta_2 - 1} \cdot (1 + \beta_3)^{\text{year}} \tag{21}$$

where F is the cumulative number of the first unit in the lot and L is the cumulative number of the last unit in the lot. The parameter value β_0 is usually called the T_1, the theoretical first unit cost. The parameters β_1 and β_2 are exponents corresponding to the learning and rate curve slopes respectively. The β_3 term is the escalation parameter.

Unlike the previous airborne radio example problem, a cost improvement curve problem generally uses a collection of data from a single program from distinct points in time. In contrast to the learning parameter, which models how prices decrease over time due to production efficiencies, the escalation parameter models how prices increase over time due to market and currency effects. Under certain production profiles, the regression analysis may not be able to discriminate between the influences of each parameter. Given a relatively flat pricing curve, this could lead to either a steep learning curve coupled with a higher escalation rate or a flatter learning curve with low escalation parameters. In either case, the system Jacobian condition number and the parameter correlation matrix can identify potential issues such as these.

As in the previous example, a multiplicative error is assumed and nonlinear regression is used to determine the parameter values.

4.1 Parameter solution development

The derivation of the nonlinear regression solution method is similar to the previous example. Due to the multiplicative error terms and through the use of the chain rule, the partial derivatives of the error function are based on the partial derivatives of the CER (14). From the definition of the CER (21), we have the list of partial derivatives as

$$\frac{\partial}{\partial \beta_0} f(\mathbf{x}, \boldsymbol{\beta}) = \left(\frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} - \left(F - \frac{1}{2}\right)^{\beta_1+1}}{\beta_1 + 1} \right) \cdot (L - (F-1))^{\beta_2 - 1} \cdot (1 + \beta_3)^{\text{year}} \tag{22}$$

$$\begin{aligned} \frac{\partial}{\partial \beta_1} f(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 \Bigg(& \frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} \cdot \ln\left(L + \frac{1}{2}\right) - \left(F - \frac{1}{2}\right)^{\beta_1+1} \cdot \ln\left(F - \frac{1}{2}\right)}{\beta_1 + 1} \\ & - \frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} - \left(F - \frac{1}{2}\right)^{\beta_1+1}}{(\beta_1 + 1)^2} \Bigg) \\ & \cdot (L - (F-1))^{\beta_2 - 1} \cdot (1 + \beta_3)^{\text{year}} \end{aligned} \tag{23}$$

$$\frac{\partial}{\partial \beta_2} f(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 \left(\frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} - \left(F - \frac{1}{2}\right)^{\beta_1+1}}{\beta_1 + 1} \right) \cdot (L - (F-1))^{\beta_2 - 1} \cdot \ln(L - (F-1)) \cdot (1 + \beta_3)^{\text{year}} \tag{24}$$

$$\frac{\partial}{\partial \beta_3} f(\mathbf{x}, \boldsymbol{\beta}) = \beta_0 \left(\frac{\left(L + \frac{1}{2}\right)^{\beta_1+1} - \left(F - \frac{1}{2}\right)^{\beta_1+1}}{\beta_1 + 1} \right) \cdot (L - (F-1))^{\beta_2 - 1} \cdot (1+\beta_3)^{\text{year}-1} \cdot \text{year} \quad (25)$$

Now with any parameter vector $\boldsymbol{\beta}^{(k)}$, the partial derivatives of the CER (22)-(25) and (14), a new parameter vector $\boldsymbol{\beta}^{(k+1)}$ can be computed from (11).

4.2 Verification and Test

The independent variable data in this application is all notional. Perfect data was generated using the parameter values listed in Table 6 and the complete data set is shown in Table 7. Multiplicative errors were randomly sampled from a normal distribution with $\mu = 0$ and $\sigma = 5\%$.

Parameter	Value
β_0	100
β_1	−0.120
β_2	−0.201
β_3	0.018

Table 6: True Cost Improvement Curve Dataset Parameters

Lot	F	L	Yr	True Cost	Err %	Noisy Cost
1	6	15	0	47.71	-3.2%	46.18
2	16	35	1	37.91	2.0%	38.66
3	36	85	2	28.95	-2.4%	28.24
4	86	185	3	23.25	-7.4%	21.53
5	186	385	4	18.82	3.9%	19.55
6	386	585	5	17.94	5.1%	18.87
7	586	785	6	17.51	-4.5%	16.72
8	786	985	7	17.28	4.0%	17.98
9	986	1185	8	17.16	0.2%	17.21
10	1186	1385	9	17.12	2.4%	17.53

Table 7: Cost Improvement Curve Example Dataset

In Table 8, the list of initial and final values are shown. The final solution was obtained after 75 iterations. Figure 7 shows the parameter values over all iterations and Figure 8 shows the parameter differences between iterations. From these plots, it is evident that after about 10 iterations the parameter values are close to the final values and close to the true parameters.

Parameter	Initial Value	Final Value	Final Error	Relative Errors
β_0	50	100	-8.29e-09	8.29e-11
β_1	-0.152	-0.120	1.03e-10	8.59e-10
β_2	-0.152	-0.201	-9.38e-11	4.67e-10
β_3	0	0.018	-1.48e-11	8.24e-10

Table 8: Computed Parameters From Cost Improvement Curve Dataset Without Error

In Figure 9, the condition number of the error function Jacobian for each iteration is plotted. The condition number of the final iteration Jacobian is around 7,600, however the initial iteration

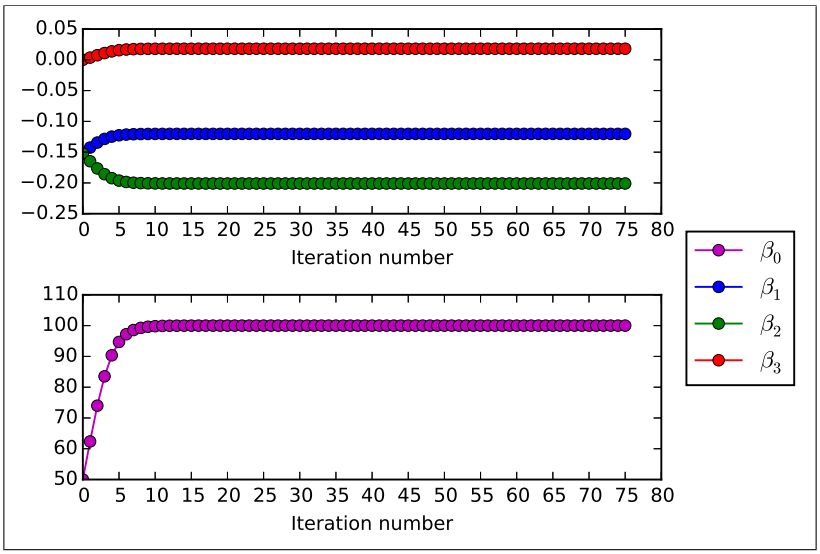

Figure 7: Computed Parameter Values by Iteration

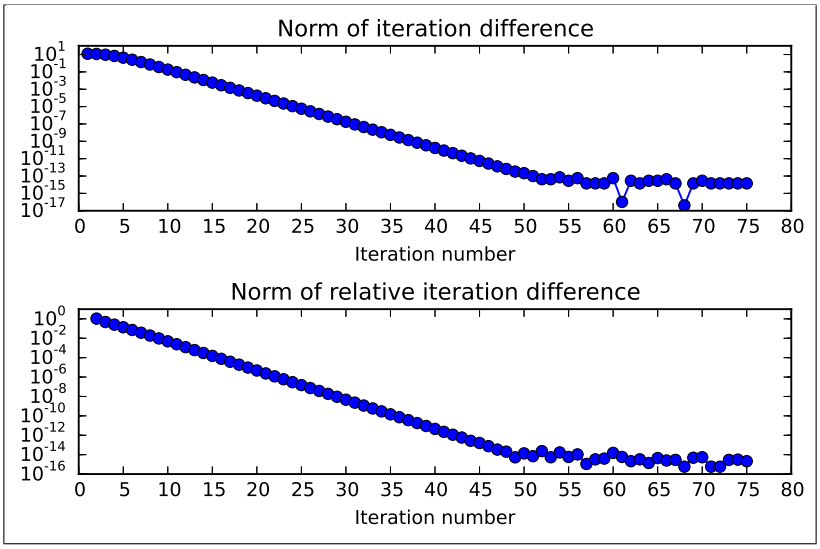

Figure 8: Computed Parameter Difference by Iteration

condition number is only about 4,000 indicating that the initial parameter vector guess produced a better conditioned system. Still, the method produces sufficiently close estimates of the true parameters.

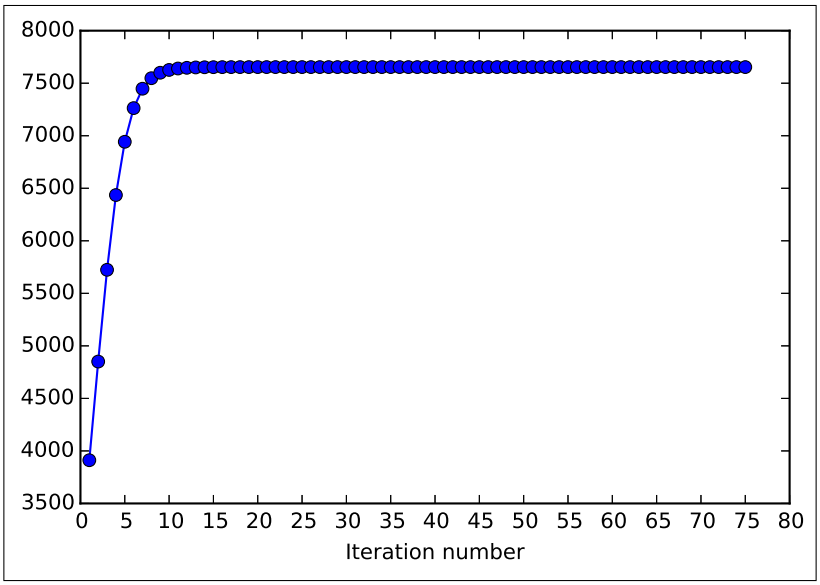

Figure 9: Jacobian Condition Number by Iteration

In Table 9, the initial guess and final computed parameter values are shown when using the Noisy Cost data. Figure 10 shows the parameter values over all iterations and Figure 11 shows the parameter differences between iterations. Again, after about 10 iterations the computed parameter values do not change substantially.

In Figure 12, the condition number of the error function Jacobian for each iteration is plotted. The condition number of the initial iteration condition number is around 4,000 and final iteration Jacobian is around 7,300. Due to the multiplicative error term (14), the Jacobian and consequently the condition number depend on the data set. Thus, while the condition numbers are similar between the perfect data problem and the data set with error, they are not exactly the same. Due to the errors in the cost data and the conditioning, the final parameters have a non-negligible difference. Specifically, the escalation parameter has the largest overall error, about 24% compared to the true parameter value.

Parameter	Initial Value	Final Value	Final Error	Relative Errors
β_0	50	95.05	4.95e+00	4.95e-02
β_1	-0.152	-0.096	-2.48e-02	2.06e-01
β_2	-0.152	-0.214	1.35e-02	6.70e-02
β_3	0	0.014	4.37e-03	2.43e-01

Table 9: Computed Parameters From Cost Improvement Curve Dataset With Error

In Table 10, the final parameter correlation matrix is shown. Here, there is a very strong correlation that exists between the learning and rate curve exponent parameters and between the learning, rate and escalation parameters.

It is worth noting, when using the perfect data the correlation matrix is almost exactly the same as the parameter correlation matrix when using the noisy cost data. The nonlinear regression

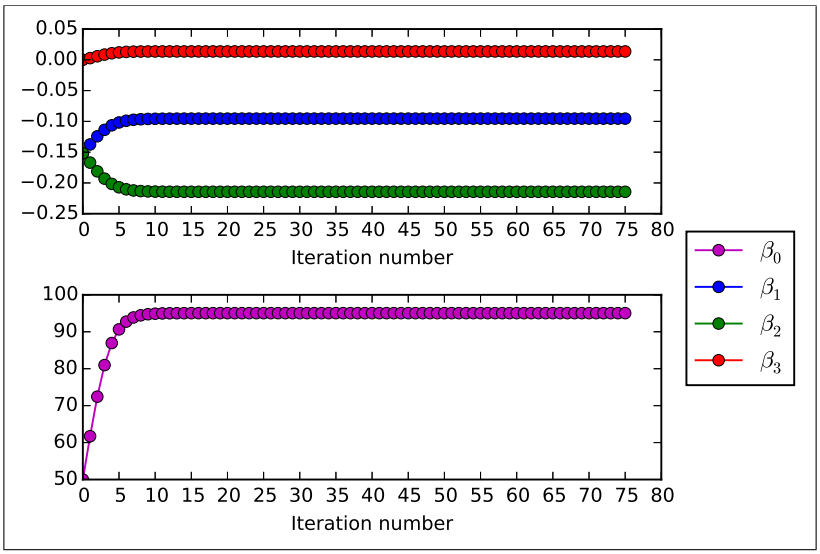

Figure 10: Computed Parameter Values by Iteration

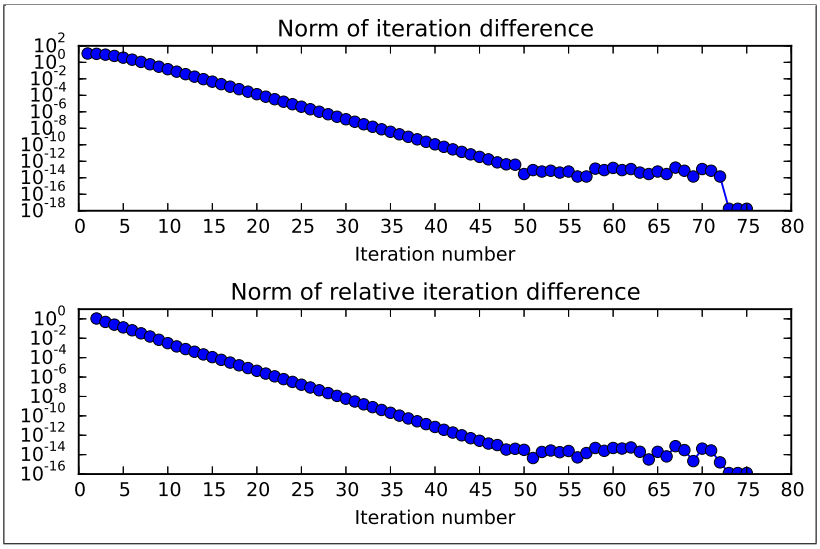

Figure 11: Computed Parameter Difference by Iteration

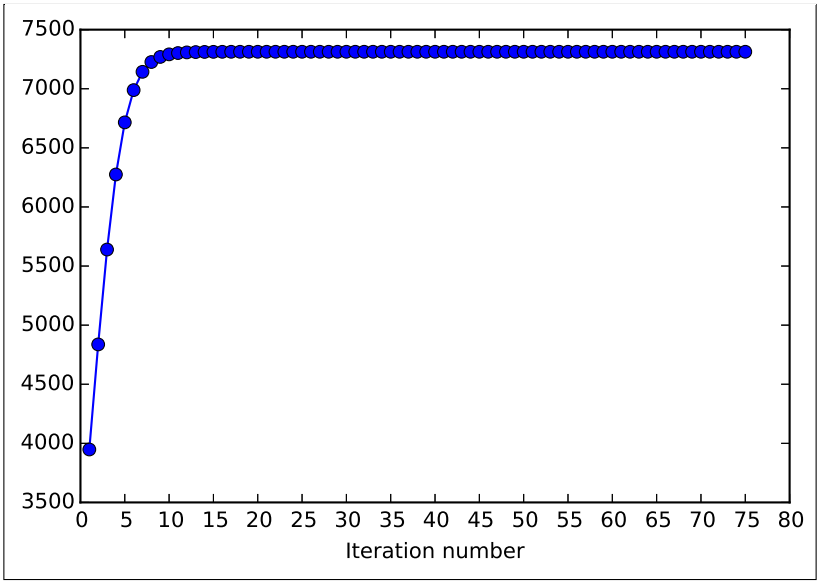

Figure 12: Jacobian Condition Number by Iteration

	β_0	β_1	β_2	β_3
β_0	1.00			
β_1	-0.726	1.00		
β_2	0.558	-0.973	1.00	
β_3	0.807	-0.978	0.912	1.00

Table 10: Parameter Correlation Matrix for Cost Improvement Curve Dataset With Error

in that case computes parameters that are very close to the true values. When the data with error is used, the nonlinear regression computes parameters with up to 25% relative error. This is considerably better than the worst case error bound provided by the condition number of over $700,000\%$ error. On the other hand, there are trivial examples of systems where the relative error of computed examples is equal to the condition number upper bound.

5 Conclusions

5.1 Summary of Results

The objective of this paper is to highlight the critical normalization step of the analysis process. In many cost estimating applications, prior to analysis, the data are normalized using general factors such as inflation or escalation indices. These normalization factors are used to homogenize the data, but can actually introduce additional errors into the data set if the factors are inappropriate for the specific data. By augmenting the model with additional parameters, the appropriate specific factors may be inferred from the data itself.

To investigate this hypothesis two typical CERs have been modified to include an escalation term. First, the model solution approach using nonlinear regression was developed. Next, using notional data, the overall numerical conditioning was analyzed when using both perfect data and data with errors. The conditioning is a measure of the multicollinearity of the independent variables, and in both applications is significantly higher than the rule of thumb warning value of 30. Despite

this, in both CER applications the method converges to the correct parameters when using perfect data. When error is present in the data, the results are usable even though they were not perfect.

An additional benefit comes from solving for the normalization factors from the data set. The analysis process can provide information regarding not just the estimated normalization parameter value but also its distribution and relationship to other parameter values. This diagnostic information can aid in quantifying the uncertainty of any prediction made from resulting models.

5.2 Future Work

While this paper does not introduce new theory but rather focuses on a potentially uncommon application of existing theory, there still remains many potential topics for future research.

As stated previously, there is considerable discussion regarding the "right" set of economic normalization indices for any particular commodity within the DoD. With larger historical data sets, analysis can be accomplished using the approach presented here to determine commodity specific normalization rates.

Additionally, since regression analysis provides information about the distribution of the parameters in addition to expected values, further research could also be focused on characterizing the impact of the overall model conditioning to the size of the joint confidence regions for model parameters for specified confidence levels. Simple heuristics or 'rules of thumb' could be identified that would improve an analysts ability to make selections between various model options.

References

[1] Air Force Instruction 65-502. 2015.

[2] Gregory A Wise, Charles B Lochbryn, and David J Oprisu. *DoD Inflation Handbook*. MCR Federal LLC, 2nd edition, 2011.

[3] Norman R Draper and Harry Smith. *Applied Regression Analysis*. John Wiley & Sons, 1966.

[4] Douglas M Bates and Donald G Watts. *Nonlinear Regression Analysis and Its Applications*. John Wiley & Sons, 1988.

[5] Joint Agency Cost Schedule Risk and Uncertainty Handbook. 2014.

[6] David Ronald Kincaid and Elliott Ward Cheney. *Numerical Analysis: Mathematics of Scientific Computing*, volume 2. American Mathematical Soc., 2002.

[7] Azmy S Ackleh, Edward James Allen, R Baker Kearfott, and Padmanabhan Seshaiyer. *Classical and Modern Numerical Analysis: Theory, Methods and Practice*. CRC Press, 2009.