Fitting Curves to Data,
Generalized Linear Least Squares, and Error Analysis

CEE 629. System Identification

Duke University, Fall 2017

It is sometimes of use to fit a curve to measured data, to determine how measurement errors propagate to errors in the curve-fit, and to quantify the confidence in the predictive capabilities of a curve-fit model.

This hand-out addresses the errors in parameters estimated from fitting a function to data. All samples of measured data include some amount of measurement error and some natural variability. Results from calculations based on measured data will also depend, in part, on random measurement errors. In other words, the normal variation in measured data propagates through calculations applied to the data.

Many (many) quantities can not be measured directly, but can be inferred from measurements: gravitational acceleration, the universal gravitational constant, seasonal rainfall into a watershed, the resistivity of a material, the elasticity of a material, metabolic rates, etc. etc. ... The Earth's gravitational acceleration can be estimated from measurements of the time it takes for a mass to fall from rest through a measured distance. The equation is $d = gt^2/2$ or $g = 2dt^{-2}$. The ruler we use to measure the distance d will have a finite resolution and may also produce systematic errors if we do not account for issues such as thermal expansion. The clock we use to measure the time t will also have some error. Fortunately, the errors associated with the ruler are in no way related to the errors in the clock. Not only are they *uncorrelated* they are *statistically independent.* If we repeat the experiment n times, with a very precise clock, we will naturally find that measurements of the time t_i are never repeated exactly. The variability within our sample of measurements of d and t will surely lead to variability in the estimation of g. As described in the next section, error propagation formulas help us determine the variability of g based upon the variability of the measurements of d and t.

This document shows how to estimate model parameters and how to evaluate the uncertainty in these parameter estimates, (such as gravitational acceleration, g) . In doing so, we will consider each individual measurement to be a random variable, in which the randomness of the variable represents the measurement error. It is reasonable (and sometimes extremely convenient) to assume that these variables are normally distributed. It is often necessary to assume that the measurement error in one measurement is statistically independent of the measurement error in all other measurements.

Estimates of the constants (i.e., fit parameters) in an equation that passes through some data points is a function of the random sample of data. So this document starts by considering the statistics (mean, standard deviation) of a function of several random variables. When reading this, think of the function as representing the coefficients in the curve-fit, and the set of random variables as the sample of measured data.

1 Quadrature Error Propagation

Given a function of several random variables, $F(Z_1, Z_2, Z_3, \cdots, Z_n)$, known variations in the independent variables, $\delta Z_1, \cdots, \delta Z_n$ will result in a computable variation δF. For small variations in Z_i we can expand the variation in F in terms of individual variations of Z_i using a Taylor series that stops at the first term and the chain rule of calculus,

$$\begin{aligned} \delta F &= \frac{\partial F}{\partial Z_1}\delta Z_1 + \frac{\partial F}{\partial Z_2}\delta Z_2 + \cdots + \frac{\partial F}{\partial Z_n}\delta Z_n. \qquad (1) \\ &= \left\{ \frac{\partial F}{\partial Z_1} \; \frac{\partial F}{\partial Z_2} \cdots \frac{\partial F}{\partial Z_n} \right\} \cdot \{\delta Z_1 \; \delta Z_2 \cdots \delta Z_n\} \\ &= \nabla F \cdot \delta \mathbf{Z} \end{aligned}$$

The notation "∇F" represents the set of derivatives of F with respect to $Z_1, \cdots, Z_n$ and is called the *gradient* of F with respect to $Z_1, Z_2, ..., Z_n$. When working with a sample of measured data, we can not know the values of the individual measurement errors (by the definition of a *measurement error*). We can't even know if the error in a measurement is positive or negative. But we can usually be satisfied by estimating only the *magnitude* of error in F, and not the sign. To do so we can estimate the square of δF. Note that the variation δF is a weighted sum of the individual measurement errors δZ_i. Assuming that the measurement errors are independent (at least for the time being) we can estimate the square of δF as

$$(\delta F)^2 = \left(\frac{\partial F}{\partial Z_1}\delta Z_1\right)^2 + \left(\frac{\partial F}{\partial Z_2}\delta Z_2\right)^2 + \cdots + \left(\frac{\partial F}{\partial Z_n}\delta Z_n\right)^2 . \qquad (2)$$

Indeed, the variance of the function is computed analogously,

$$\sigma_F^2 = \left(\frac{\partial F}{\partial Z_1}\right)^2 \sigma_{Z_1}^2 + \left(\frac{\partial F}{\partial Z_2}\right)^2 \sigma_{Z_2}^2 + \cdots + \left(\frac{\partial F}{\partial Z_n}\right)^2 \sigma_{Z_n}^2 = \sum_{i=1}^{m} \left(\frac{\partial F}{\partial Z_i}\right)^2 \sigma_{Z_i}^2$$

The table below illustrates that in some special cases the error in the function may be conveniently simplified, while in other cases it cannot.

function	**error propagation formula**
$F = \sum_{i=1}^{n} Z_i$	$(\delta F)^2 = \sum_{i=1}^{n} (\delta Z_i)^2$
$F = aZ_1 + bZ_2 - cZ_3$	$(\delta F)^2 = (a\delta Z_1)^2 + (b\delta Z_2)^2 + (c\delta Z_3)^2$
$F = aZ_1Z_2/Z_3$	$\left(\frac{\delta F}{F}\right)^2 = \left(\frac{\delta Z_1}{Z_1}\right)^2 + \left(\frac{\delta Z_2}{Z_2}\right)^2 + \left(\frac{\delta Z_3}{Z_3}\right)^2$
$F = aZ^p$	$\left\lvert\frac{\delta F}{F}\right\rvert = \left\lvert p\frac{\delta Z}{Z}\right\rvert$
$F = aZ_1^p - bZ_2^q$	$(\delta F)^2 = (apZ_1^{p-1}\delta Z_1)^2 + (bqZ_2^{q-1}\delta Z_2)^2$

If the random variables Z are correlated, with a covariance matrix V_Z, then the variance of the function $F(Z)$ is

$$\begin{aligned} \sigma_F^2 &= \sum_{i=1}^{m}\sum_{j=1}^{m} \frac{\partial F}{\partial Z_i}\frac{\partial F}{\partial Z_j}[\mathsf{V}_Z]_{i,j} \\ &= \left(\frac{\partial F}{\partial Z}\right) \mathsf{V}_Z \left(\frac{\partial F}{\partial Z}\right)^{\mathsf{T}} , \qquad (3) \end{aligned}$$

where $(\partial F/\partial Z)$ is the m-dimensional row-vector of the gradient of F with respect to Z, and $[\mathsf{V}_Z]_{i,i} = \sigma^2_{Z_i}$. Finally, if $F(Z)$ is an m-dimensional vector-valued function of n correlated random variables, with covariance matrix V_Z, then the $m \times m$ covariance matrix of F is

$$\begin{aligned} [\mathsf{V}_F]_{k,l} &= \sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial F_k}{\partial Z_i}\frac{\partial F_l}{\partial Z_j}[\mathsf{V}_Z]_{i,j} \\ \mathsf{V}_F &= \left[\frac{\partial F}{\partial Z}\right] \mathsf{V}_Z \left[\frac{\partial F}{\partial Z}\right]^{\mathsf{T}} . \end{aligned} \tag{4}$$

where $[\partial F/\partial Z]$ is an $m \times n$ matrix and is called the *Jacobian* of F with respect to Z. The *Jacobian* quantifies the *sensitivity* of each element F_k to each Z_i individually,

$$\left[\frac{\partial F}{\partial Z}\right]_{k,i} = \frac{\partial F_k}{\partial Z_i}, \tag{5}$$

and the *covariance* quantifies the variability amongst the data. Equation (4) shows how the sensitivities of a function to its variables and the variability amongst the variables themselves are combined in order to estimate the variability of the function. It is centrally important to what follows.

2 An Example of Linear Least Squares

Measured data are often used to estimate the coefficients of an equation or the parameters of a model. Curve-fitting is a common example of this. Consider the fitting of a function $\hat{y}(x; a)$ that involves a set of coefficients $a_1, ...a_n$, to a set of m measured data points (x_i, y_i), $i = 1, ..., m$. If the function is linear in the coefficients then the relationship between the data and the coefficients can always be written as a matrix-vector product

$$\hat{y}(x; a) = Xa$$

where

- a is the vector of the coefficients to be estimated.
- $\hat{y}(x; a)$ is the function to be fit to the m data coordinates (y_i, x_i), and
- the matrix X depends only on the set of independent variables, x.

As a general example, consider the problem of fitting an $(n - 1)$ degree power-polynomial to m measured data coordinates, (x_i, y_i), $i = 1, \cdots, m$. The general form of the equation is

$$\hat{y}(x; a) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{n-1} x^{n-1}, \tag{6}$$

The equation may be written m times for every data coordinate (x_i, y_i), $i = 1, \cdots, m$,

$$\begin{aligned} \hat{y}_1(x, a) &= a_0 + a_1 x_1 + a_2 x_1^2 + \quad \cdots + a_{n-1} x_1^{n-1} \\ \hat{y}_2(x, a) &= a_0 + a_1 x_2 + a_2 x_2^2 + \quad \cdots + a_{n-1} x_2^{n-1} \\ \vdots & \qquad\qquad\qquad\qquad\quad \ddots \\ \hat{y}_m(x, a) &= a_0 + a_1 x_m + a_2 x_m^2 + \quad \cdots + a_{n-1} x_m^{n-1} \end{aligned}$$

or, in matrix form as $\hat{y}(x;a) = Xa$,

$$\left\{ \begin{array}{c} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_m \end{array} \right\} = \left[\begin{array}{ccccc} 1 & x_1 & x_1^2 & \cdots & x_1^{n-1} \\ 1 & x_2 & x_2^2 & \cdots & x_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_m & x_m^2 & \cdots & x_m^{n-1} \end{array} \right] \left\{ \begin{array}{c} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_{n-1} \end{array} \right\} . \qquad (7)$$

Matrices structured in the form of X in equation (7) are called vanderMonde and arise in cuve-fitting problems. The number of parameters (n in this example) must always be less than or equal to the number of data coordinates, m. We can assume that each measurement y_i has a particular variance, $\sigma^2_{y_i}$, and, unless we know better, we should further assume that the sample of measurements y_i are uncorrelated.

Fitting the equation to the data reduces to estimating values of n parameters, $a_0, \cdots, a_{n-1}$ such that the equation represents the data *as closely as possible*.

But what does "*as closely as possible*" even mean?

In 1829 Carl Friedrich Gauss proved that it is physically sound *and* mathematically convenient to say that the best-fit curve minimizes the sum of the squares of the errors between the data and the curve. So to find estimates for the parameters, we would like to minimize the sum of the squares of the residual errors r_i between the data, y_i, and the equation, $\hat{y}(x_i;a)$.

$$r_i = \hat{y}(x_i;a) - y_i .$$

Furthermore, the error should be more heavily weighted with the high-accuracy (small-variance) data and less heavily weighted with the low-accuracy (large-variance) data. This curve fit criterion is called the *weighted sum of the squared residuals* (WSSR), also called the 'chi-squared' error function,

$$\chi^2(a) = \sum_{i=1}^{m} \left[\hat{y}(x_i;a) - y_i \right]^2 / \sigma^2_{y_i} \qquad (8)$$

An equivalent expression for $\chi^2(a)$ may be written in terms of the vectors y and a, the system matrix X, and the data covariance matrix V_y

$$\begin{aligned} \chi^2(a) &= \{Xa - y\}^{\mathsf{T}} \mathsf{V}_y^{-1} \{Xa - y\}, \\ &= a^{\mathsf{T}} X^{\mathsf{T}} \mathsf{V}_y^{-1} X a - 2 a^{\mathsf{T}} X^{\mathsf{T}} \mathsf{V}_y^{-1} y + y^{\mathsf{T}} \mathsf{V}_y^{-1} y . \qquad (9) \end{aligned}$$

Prove to yourself that if V_y is the diagonal matrix of variances of y, $\sigma^2_{y_i}$, then equation (9) is equivalent to equation (8). If the data covariance matrix is not diagonal, then equation (9) is a generalization of (8).

Note that any weighted least squares problem can be scaled to an unweighted least squares problem as long as the weighting matrix is symmetric and positive-definite. Decomposing the weighting matrix into Cholesky factors, $\mathsf{V}_y^{-1} = R^{\mathsf{T}} R$, and defining $\bar{y} = Ry$ and $\bar{X} = RX$, any weighted criterion (9) is equivalent to the unweighted criterion, with no loss of generality.

$$\chi^2(a) = \{\bar{X}a - \bar{y}\}^{\mathsf{T}} \{\bar{X}a - \bar{y}\} .$$

3 Complete Solution to a Linear Least-Squares problem

The elements of a complete parameter estimation analysis are:

1. the numerical values of the parameter estimates, $\hat{a}$ that minimize an error criterion;
2. the function or model evaluated with the parameter estimates, $\hat{y}(x; \hat{a})$;
3. a value for the goodness-of-fit criterion;
4. the parameter covariance matrix, $\mathsf{V}_{\hat{a}}$;
5. the standard error of the parameters, $\sigma_{\hat{a}}$; and
6. the standard error of the fit, $\sigma_{\hat{y}}$.

3.1 The least-squares fit parameters and the best-fit curve

The χ^2 error function is minimized with respect to the parameters a, by solving the system of equations

$$\begin{aligned} \left.\frac{\partial \chi^2}{\partial a}\right|_{a=\hat{a}} &= 0 \\ X^{\mathsf{T}} \mathsf{V}_y^{-1} X\hat{a} - X^{\mathsf{T}} \mathsf{V}_y^{-1} y &= 0 \\ \hat{a} &= [X^{\mathsf{T}} \mathsf{V}_y^{-1} X]^{-1} X^{\mathsf{T}} \mathsf{V}_y^{-1} y \qquad (10) \end{aligned}$$

Equation (10) provides the ordinary linear least square estimate of the parameters from the measured data, y, the data covariance matrix V_y, and the system matrix, X. In many cases we do not know the standard deviation of each measurement error individually, (or their correlations). So it is common to assume that every measurement has the same distribution of measurement errors and that the measurement errors are uncorrelated. With these assumptions, $\mathsf{V}_y = \sigma_y^2 I$. Substituting this into equation (10), we find that in the case of equal and uncorrelated measurement error, the parameter estimates are independent of the measurement error.

$$\begin{aligned} \hat{a} &= \left[X^{\mathsf{T}} \frac{1}{\sigma_y^2} X\right]^{-1} X^{\mathsf{T}} \frac{1}{\sigma_y^2} y \\ \hat{a} &= \left[X^{\mathsf{T}} X\right]^{-1} X^{\mathsf{T}} y \ . \qquad (11) \end{aligned}$$

Once the parameter estimates are found, the best fit curve is easily computed using

$$\hat{y}(x; \hat{a}) = X\hat{a} = X\left[X^{\mathsf{T}} X\right]^{-1} X^{\mathsf{T}} y, \qquad (12)$$

which is a projection of the data y onto the space of models $\hat{y} = X\hat{a} = \Pi_X \ y$.

3.2 Goodness of fit

Because χ^2 is the error criterion, the value of χ^2 quantifies of the quality of the fit. The value of χ^2 can be normalized to a value that is more broadly meaningful. The unbiased estimate of the variance of the weighted residuals is $\chi^2/(m-n+1)$. In the absence of any other information on the variance of the data, V_y, χ^2 can be computed without weighting the residuals, i.e., using $\sigma_{y_i}=1$. In this case the variance of the measurement error can be estimated as the variance of the unweighted residuals,

$$\sigma_y^2 = \frac{\chi^2}{m-n+1} \ , \tag{13}$$

where m is the number of measurement values and n is the number of parameters. This estimate of the variance of the residuals can be interpreted as the measurement error if the resulting residuals are Gaussian and uncorrelated.

The R^2 goodness-of-fit criterion compares the variability in the measurements not explained by the model to the total variability in the measurements.

$$R^2 = 1 - \frac{\sum[y_i - \hat{y}(x_i;\hat{a}]^2}{\sum[y_i-\bar{y}]^2} \ , \tag{14}$$

where $\bar{y}$ is the average value of the measured data values. The R^2 criterion is the ratio of the variability in the data that is not explained by the model to the total variability in the data. A value of $R^2=0$ means that the model does not explain the measurement variability any better than the mean measurement value; a negative value of R^2 means that the model explains the measurement variability worse than the mean measurement value, and a value of $R^2=1$ means that all of the variability in the data is fully explained by the model, i.e., there is no unexplained measurement variability.

3.3 The Covariance and standard error of the parameters

The covariance matrix of the parameter estimates, $V_{\hat{a}}$, is a direct application of equation (4)

$$\begin{aligned} \mathsf{V}_{\hat{a}} &= \left[\frac{\partial \hat{a}}{\partial y}\right] \mathsf{V}_y \left[\frac{\partial \hat{a}}{\partial y}\right]^{\mathsf{T}} \\ &= [X^{\mathsf{T}}\mathsf{V}_y^{-1}X]^{-1}X^{\mathsf{T}}\mathsf{V}_y^{-1}\mathsf{V}_y\mathsf{V}_y^{-1}X[X^{\mathsf{T}}\mathsf{V}_y^{-1}X]^{-1} \\ &= [X^{\mathsf{T}}\mathsf{V}_y^{-1}X]^{-1}[X^{\mathsf{T}}\mathsf{V}_y^{-1}X][X^{\mathsf{T}}\mathsf{V}_y^{-1}X]^{-1} \\ &= [X^{\mathsf{T}}\mathsf{V}_y^{-1}X]^{-1} \ . \end{aligned} \tag{15}$$

This covariance matrix is sometimes called the error propagation matrix, as it indicates how random measurement errors in the data y, as described by V_y, propagate to the model coefficients $\hat{a}$. If no prior information regarding the measurement error covariance is available, and χ^2 is computed without weighting the residuals, then the parameter covariance matrix may be calculated from the expression

$$\mathsf{V}_{\hat{a}} = \frac{\chi^2}{m-n}[X^{\mathsf{T}}X]^{-1} \ . \tag{16}$$

The standard error of the parameters, $\sigma_{\hat{a}}$, is the square-root of the diagonal of the parameter covariance matrix. The standard error of the parameters is used to determine confidence intervals for the parameters,

$$\hat{a} - t_{1-\alpha/2}\, \sigma_{\hat{a}} \;\le\; a \;\le\; \hat{a} + t_{1-\alpha/2}\, \sigma_{\hat{a}}, \tag{17}$$

where $1-\alpha$ is the desired confidence level, and t is the Student-t statistic. When the number of measurement values is much larger than the number of estimated parameters ($m-n > 100$), use $t = 1.96$ for 95% confidence intervals and $t = 1.645$ for 90% confidence intervals, otherwise the value of t will depend also on $(m-n)$.

3.4 The standard error of the fit

As in the computation of the parameter covariance, the covariance of the fit, $\mathsf{V}_{\hat{y}}$, is a direct application of equation (4). The variability in the fit is due to variability in the parameters, $\mathsf{V}_{\hat{a}}$, which, in turn is due to variability in the data. Again applying equation (4),

$$\mathsf{V}_{\hat{y}} = \left[\frac{\partial \hat{y}}{\partial a}\right] \mathsf{V}_{\hat{a}} \left[\frac{\partial \hat{y}}{\partial a}\right]^{\mathsf{T}}, \tag{18}$$

$$= X\, \mathsf{V}_{\hat{a}}\, X^{\mathsf{T}}. \tag{19}$$

Note that $\mathsf{V}_{\hat{y}}$ is an m-by-m matrix. The standard error of the fit, $\sigma_{\hat{y}}$, is the square-root of the diagonal of $\mathsf{V}_{\hat{y}}$, and is used to determine confidence intervals for the fit.

$$\hat{y}(x;\hat{a}) - t_{1-\alpha/2}\, \sigma_{\hat{y}} \;\le\; y \;\le\; \hat{y}(x;\hat{a}) + t_{1-\alpha/2}\, \sigma_{\hat{y}}. \tag{20}$$

The standard prediction error must account for both the standard error of the fit and the variability of the data.

$$\mathsf{V}_{\hat{y}\mathsf{p}} = \mathsf{V}_{\hat{y}} + \mathsf{V}_{y} = X\, \mathsf{V}_{\hat{a}}\, X^{\mathsf{T}} + \sigma_y^2 \tag{21}$$

The standard prediction error, $\sigma_{\hat{y}\mathsf{p}}$, is the square-root of the diagonal of $\mathsf{V}_{\hat{y}\mathsf{p}}$.

4 Implementation — mypolyfit.m

```
function [a,y_fit,Sa,Sy_fit,R2,Ca,condX] = mypolyfit(x,y,p,b,Sy)
% [a,y_fit,Sa,Sy_fit,Ca,condX] = mypolyfit(x,y,p,b,Sy)
%
% fit a power-polynomial, y_fit(x;a) to data pairs (x,y) where
%
%  y_fit(x;a) = SUM_i=1 ^ length(p)  a_i  x^p_i
%
% which minimizes the Chi-square error criterion, X2,
%
%  X2 = SUM_k=1 ^ length(x) [ ( y_fit(x_k;a) - y_k )^2 / Sy_k^2 ]
%
% where  Sy_k is the standard error of the k-th data point.
%
% INPUT VARIABLES
%
%   x  = vector of measured values of the independent variables,
%        Note: x is assumed to be assumed to be error-free
%   y  = corresponding values of the dependent variables
%        Note: length of y must equal length of x
%   p  = vector of powers to be included in the polynomial fit
%        Note: values of p may be any real number
%   b  = regularization constant                           default = 0
%   Sy = standard errors of the independent variables      default = 1
%
% OUTPUT VARIABLES
%
%   a      = identified values of the polynomial coefficients
%   y_fit  = values of curve-fit evaluated at values of x
%   Sa     = standard errors of polynomial coefficients
%   R2     = R-squared error criterion
%   Sy_fit = standard errors of the curve-fit
%   Ca     = parameter correlation matrix
%   condX  = condition number of system matrix
%
% Henri Gavin, Civil Engineering, Duke Univ., Durham NC   4-10-2007

% error checking

 if ( length(x) ~= length(y) )
    disp(' length of x must equal length of y ');
    return
 end

 Ny = length(y);
 Np = length(p);

 x = x(:);              % make "x" a column-vector
 y = y(:);              % make "y" a column-vector
 p = p(:);              % make "p" a column-vector

% default values

%  ... set up inverse of y data covariance matrix, P
 if nargin > 4,   P  = diag(1./Sy.^2);        else, P = eye(Ny);  end
%  ... regularization parameter
 if nargin < 4,   b = 0;                                         end

 xm = max(x);

 X = zeros(Ny,Np);      % allocate memory for "X" matrix

 for i=1:Np
     X(:,i) =  x.^p(i);                % set up X matrix such that y = X a
 end

 condX = cond( X'*P*X + b*eye(Np) );            % condition number
 C = inv( X'*P*X + b*eye(Np) );                 % parameter "covariance"
```

```
a = C * X'*P*y;                                % least squares parameters

y_fit = X*a;                                   % least squares fit

if nargin < 5                  % estimate data covariance from the curve-fit
   noise_sq = sum((y-y_fit).^2)/(Ny-Np);       % sum of squared errors
   P = eye(Ny) / noise_sq;                     % data covariance (inverse)
   C = inv( X'*P*X + b*eye(Np) );              % estimated parameter covariance
end


                       % re-compute parameter covariance for b ~= 0
if b == 0              % no regularization
   Va = C;             % simple expression for parameter covariance
else
   Va = C*(X'*P*X + b^2*C*X'*P*y*y'*P*X*C)*C;  % ... more complicated!
end

Sa   = sqrt(diag(Va)); % standard error of the parameters
Sa   = Sa(:);

                       % standard error of the curve-fit
Sy_fit = sqrt(diag(X*Va*X'));          % Vy = [dy/da] Va [dy/da]' = X Va X'

R2 = 1 - sum( (y-y_fit).^2 ) / sum( (y-sum(y)/Ny).^2 );        % R-squared

Ca = Va ./ (Sa * Sa');                 % parameter cross-correlation matrix

disp('     p         a            +/-   da           (percent) ')
disp('---------------------------------------------------------')
for i=1:Np
  if rem(p,1) == 0
     fprintf('   a[%2d] =  %11.3e;    +/- %10.3e     (%5.2f %%)\n',  ...
                               p(i), a(i), Sa(i), 100*Sa(i)/abs(a(i)) );
  else
     fprintf(' %8.2f :  %11.3e     +/- %10.3e     (%5.2f %%)\n',  ...
                               p(i), a(i), Sa(i), 100*Sa(i)/abs(a(i)) );
  end
end

% ------------------------------------------------------------------ mypolyfit
```

5 Example

Examples of fitting power polynomials to data is illustrated below. The data to be curve-fit is shown in Figures 1 and 2.

A curve fit using exponents of -1, 0, 1, 2, 3, and 4 results in the curve-fit shown in Figure 1, and the data shown below.

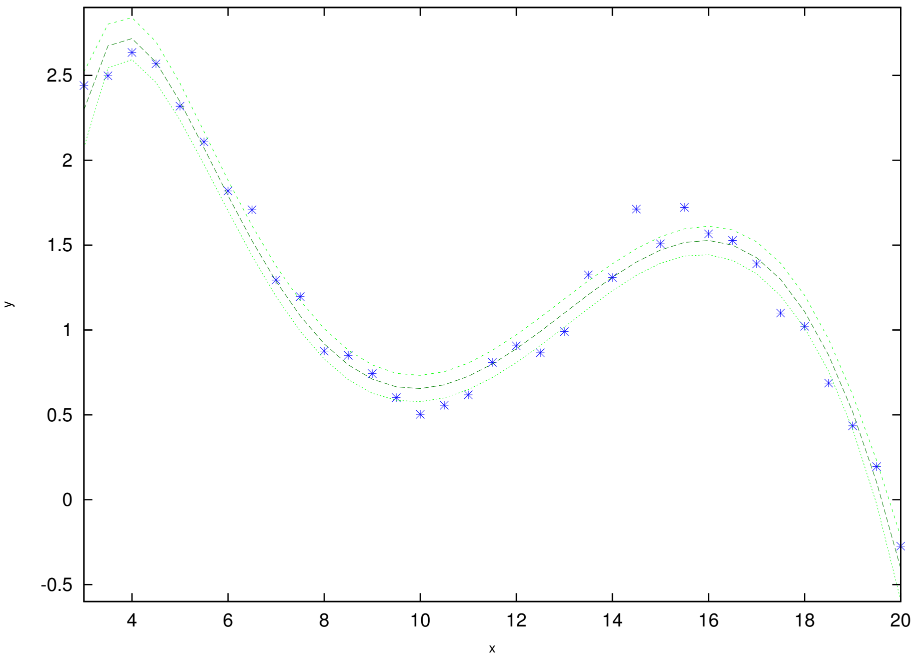

Figure 1. Example of linear least squares curve-fit, with exponents of -1, 0, 1, 2, 3, and 4

The results of mypolyfit.m are as follows:

```
     p         a            +/-   da           (percent)
---------------------------------------------------------
   a[-1] =   -3.969e+01;    +/-  6.482e+00     (16.33 %)
   a[ 0] =    2.791e+01;    +/-  4.317e+00     (15.47 %)
   a[ 1] =   -5.156e+00;    +/-  1.033e+00     (20.03 %)
   a[ 2] =    3.687e-01;    +/-  1.121e-01     (30.42 %)
   a[ 3] =   -8.340e-03;    +/-  5.600e-03     (67.15 %)
   a[ 4] =   -2.472e-05;    +/-  1.042e-04     (421.73 %)
R2 =  0.97328
```

Note the very small magnitude of the coefficient for the x^4 term, and the very large relative standard errors, not only for the x^4 coefficient (420%), but for all coefficients.

Eliminating the x^4 term from the curve-fit results in essentially the same R^2 value but with much smaller standard parameter errors.

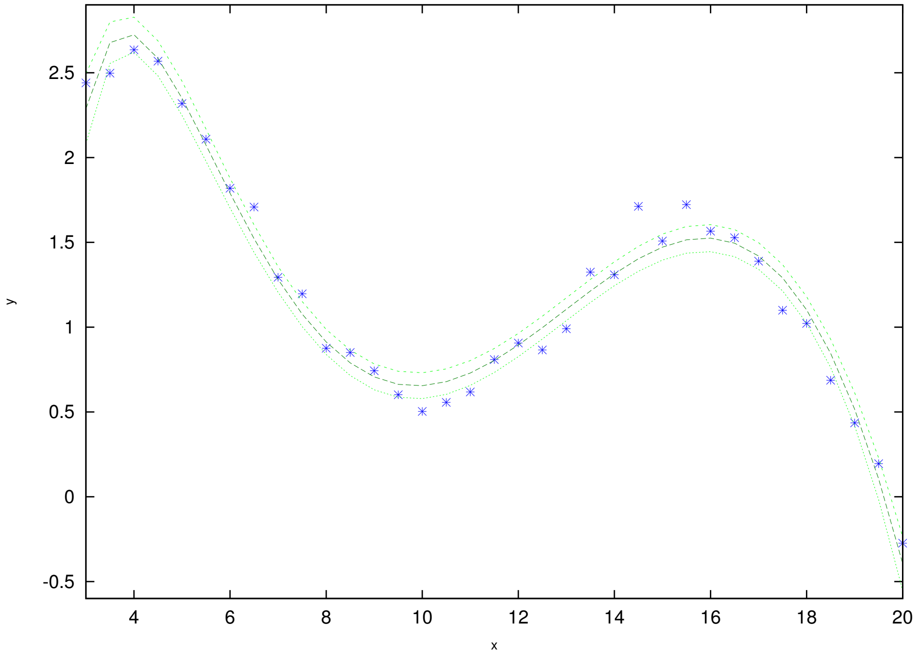

Figure 2. Example of linear least squares curve-fit, with exponents of -1, 0, 1, 2, and 3

```
     p         a             +/-   da          (percent)
---------------------------------------------------------
   a[-1] =   -4.106e+01;    +/-  2.902e+00     ( 7.07 %)
   a[ 0] =    2.886e+01;    +/-  1.549e+00     ( 5.37 %)
   a[ 1] =   -5.392e+00;    +/-  2.710e-01     ( 5.03 %)
   a[ 2] =    3.949e-01;    +/-  1.880e-02     ( 4.76 %)
   a[ 3] =   -9.663e-03;    +/-  4.453e-04     ( 4.61 %)
R2 =  0.97323
```

Comparing these two cases, one sees that in the first case, with the x^4 term, the parameter values are much more sensitive to random measurement errors than in the second case, without the x^4 term, even though the two curve-fit lines $\hat{y}(x; \hat{a})$ and the standard errors of the fits $\sigma_{\hat{y}}$ are practically indistinguishable from one another.

```
% test_mypolyfit.m
% test the mypolyfit function for least squares data fitting and error analysis

x_min =  3;              % minimum value of the indpendent values
x_max = 20;              % maximum value of the indpendent values
dx    =  0.5;            % increment of the independent variable

meaurement_error = 0.1;  % root-mean-square of the simulated measurement error

x = [ xmin : dx : xmax ]';

Nx = length(x);

randn('seed',1);        % seed the random number generator

y = 0.9*sin(x/2) + abs(x).*exp(-abs(x)/5) +  measurement_error*randn(Nx,1);

p = [ 0 1 2 3 4 5 ];            % powers involved in the fit

xs = 1;         % scale data such that max(x) = xs
b  = 0;         % regularization

[ a, y_fit, Sa, Sy_fit, R2, Ca,condX ] = mypolyfit(x,y,p);

figure(1);
 plot(x,y,'o', x,y_fit,'-', x,y_fit+1.96*Sy_fit,'--', x,y_fit-1.96*Sy_fit,'--')

R2
condX

% ------------------------------------------------ test_mypolyfit
```

6 What could possibly go wrong?

In fitting a model to data we hope for precise estimates of the model parameters, that is parameter estimates that are relatively unaffected by measurement noise. The precision of the model parameters is quantified by the parameter covariance matrix, equation (15), from which the standard error of the model parameters are computed. Large parameter covariance can result from noisy measurements, a poorly conditioned basis for the model, or both. If two or more vectors that form the columns of X (the basis of the model) are nearly linearly dependent then the estimates of the coefficients corresponding to those columns of X will be significantly affected by noise in the data, and those parameters will have broad confidence intervals. Ill-conditioned matrices have a very small determinant (for square matrices) or very large condition numbers.

These concepts are addressed in more detail in subsequent sections, which describe four methods to obtain tighter confidence intervals in the model parameters: scaling, orthogonal polynomials, regularization, and singular value decomposition.

7 Scaling to Improve Conditioning

Consider the fitting of a n degree polynomial to data over an interval $[\alpha, \beta]$. This typically involves finding the pseudo-inverse of a *VanderMonde* matrix of the form

$$X = \begin{bmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^n \\ 1 & \alpha + h & (\alpha + h)^2 & \dots & (\alpha + h)^n \\ 1 & \alpha + 2h & (\alpha + 2h)^2 & \dots & (\alpha + 2h)^n \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta - 2h & (\beta - 2h)^2 & \dots & (\beta - 2h)^n \\ 1 & \beta - h & (\beta - h)^2 & \dots & (\beta - h)^n \\ 1 & \beta & \beta^2 & \dots & \beta^n \end{bmatrix} \tag{22}$$

where the independent variable is uniformly sampled with a sample interval of h. The following table indicates the condition number of $[X^\mathsf{T} X]$ for various polynomial degrees and curve-fitting intervals.

n	$\det([X^\mathsf{T}X])$	$\det([X^\mathsf{T}X])$
	$\alpha = 0, \beta = 1$	$\alpha = -1, \beta = 1$
2	10^2	10^4
4	10^{-2}	10^3
6	10^{-11}	10^2
8	10^{-24}	10^{-2}

This table illustrates that the conditioning of the VanderMonde matrix for polynomial curve-fitting depends upon the interval over which the data is to be fit. To explore this idea further, consider the VanderMonde matrix for power polynomial curve-fitting over the domain $[-L, L]$. The condition number of X for various polynomial degrees (n) and intervals, L is shown in the figure 3. This figure illustrates that the curve-fit interval that minimizes the condition number of X depends upon the polynomial degree n. The minimum condition number is plotted with respect to the polynomial degree in figure 4, along with a curve-fit. To minimize the condition number of the VanderMonde matrix for curve-fitting a n-th degree polynomial, the curve-fit should be carried out over the domain $[-L, L]$, where $L = 1.14 + 0.62/n$. So, if we change variables before doing the curve-fit to an interval $[-L, L]$ then our results will be more accurate, i.e., less susceptible to the errors of finite precision calculations.

Consider two related polynomials for the same function

$$\hat{y}(x; a) = \sum_{k=0}^{n} a_k x^k \qquad\qquad a \le x \le b \tag{23}$$

$$\hat{y}(u; b) = \sum_{k=0}^{n} b_k u^k \qquad\qquad -L \le u \le L \tag{24}$$

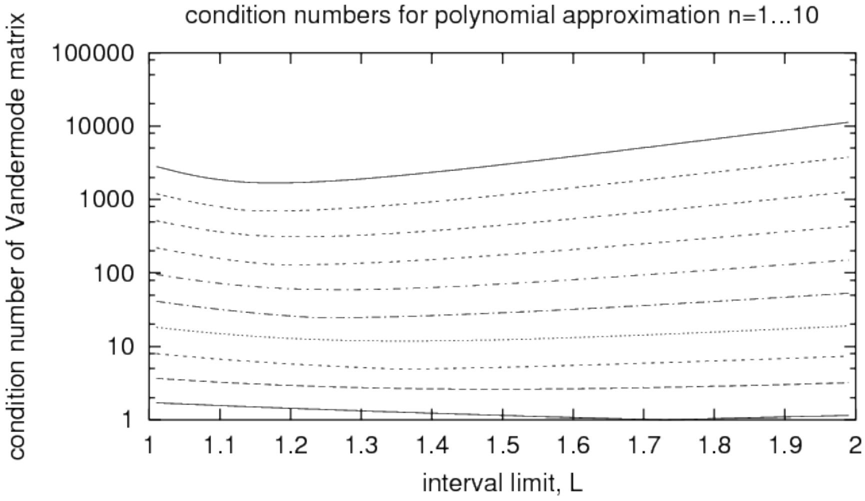

Figure 3. Condition number of VanderMonde matrices for power polynomials.

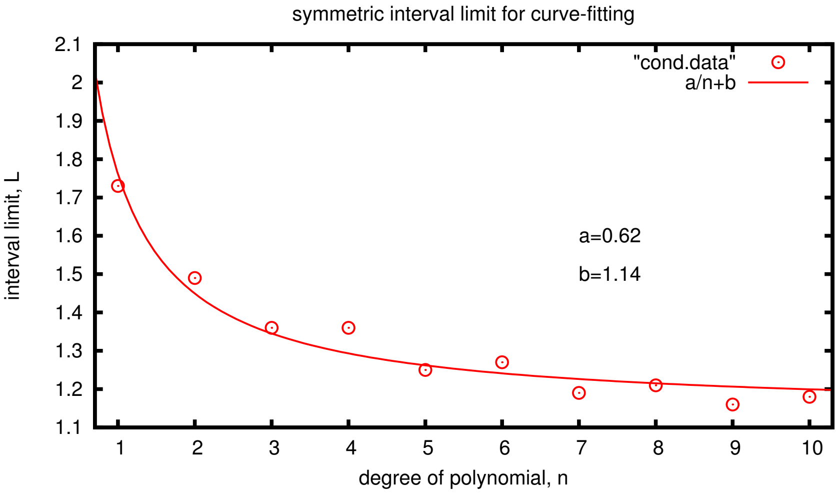

Figure 4. Scaling interval that minimizes the condition number for power polynomial fitting.

with the linear mappings

$$x = \frac{1}{2}(\alpha+\beta) + \frac{1}{2}(\alpha-\beta)\frac{u}{L} \tag{25}$$

$$u = \frac{2L}{\alpha-\beta}x + \frac{\alpha+\beta}{\beta-\alpha}L = qx + r, \tag{26}$$

then

$$y = \sum_{k=1}^{n} b_k (qx+r)^k \qquad\qquad -L \le x \le L \tag{27}$$

Solving the least squares problem for the coefficients b_k is more accurate than solving the least squares problem for the coefficients a_k. Then, given a set of coefficients b_k, along with

α and β, the coefficients a_k are recovered from

$$a_k = \sum_{j=k}^{n} b_j \frac{\prod_{i=0}^{k+1}(j-i)}{(j-k)!} \left(\frac{2L}{a-b}\right)^k \left(L\frac{a+b}{b-a}\right)^{j-k} \tag{28}$$

8 Orthogonal Polynomial Bases

The previous section shows that the ill-conditioned bases of high-order power polynomials can be partially improved by mapping the domain to $[-1:1]$.

If conditioning remains a problem after mapping the domain, the power polynomial basis can be replaced with a basis of orthogonal polynomials, for which $[X^{\mathsf{T}}WX]$ is diagonal for certain diagonal matrices W that are determined from the selected basis of orthogonal polynomials.

A set of functions, $f_i(x)$, $i = 0, \cdots, n$, is *orthogonal* with respect to some weighting function, $w(x)$ in an interval $\alpha \le x \le \beta$ if

$$\int_{\alpha}^{\beta} f_i(x)\ w(x)\ f_j(x)\ dx = \begin{cases} 0 & i \neq j \\ P_i > 0 & i = j \end{cases} \tag{29}$$

Furthermore, a set of functions is *orthonormal* if $P_i = 1$. Division by $\sqrt{P_i}$ normalizes the set of orthogonal functions, $f_i(x)$. For example, the sine and cosine functions are orthogonal.

$$\int_{-\pi}^{\pi} \cos mx\ \sin nx\ dx = 0\ \forall\ m,n \tag{30}$$

$$\int_{-\pi}^{\pi} \cos mx\ \cos nx\ dx = \begin{cases} 2\pi & m = n = 0 \\ \pi & m = n \neq 0 \\ 0 & m \neq n \end{cases} \tag{31}$$

$$\int_{-\pi}^{\pi} \sin mx\ \sin nx\ dx = \begin{cases} 2\pi & m = n = 0 \\ \pi & m = n \neq 0 \\ 0 & m \neq n \end{cases} \tag{32}$$

An orthogonal polynomial, $f_n(x)$ (of degree n) has n real distinct roots within the domain of orthogonality. The n roots of $f_n(x)$ are separated by the $n-1$ roots of $f_{n-1}(x)$. All orthogonal polynomials satisfy a recurrence relationship

$$A_{k+1}f_{k+1}(x) = B_{k+1}\ x\ f_k(x) + C_{k+1}f_k(x) + D_{k+1}f_{k-1}(x), \qquad k \ge 1 \tag{33}$$

which is often useful in generating numerical values for the polynomial basis.

8.1 Examples of Orthogonal Polynomials

- The set of Legendre polynomials, $P_k(x)$, solve the ordinary differential equation

$$\left[(1-x^2)P_k'(x)\right]' + k(k-1)P_k(x) = 0$$

and arises in solution to Laplace's equations in spherical coordinates. They are orthogonal over the domain $[-1, 1]$ with respect to a unit weight.

$$\int_{-1}^{1} P_m(x)\ P_n(x)\ dx = \begin{cases} \frac{2}{2n+1} & m = n \\ 0 & m \neq n \end{cases} \tag{34}$$

The recurrence relationship is

$$(k+1)P_{k+1}(x) = (2k+1)xP_k(x) + kP_{k-1}(x) = 0, \tag{35}$$

with $P_0(x) = 1$, $P_1(x) = x$, and $P_2(x) = (3/2)x^2 - 1/2$.

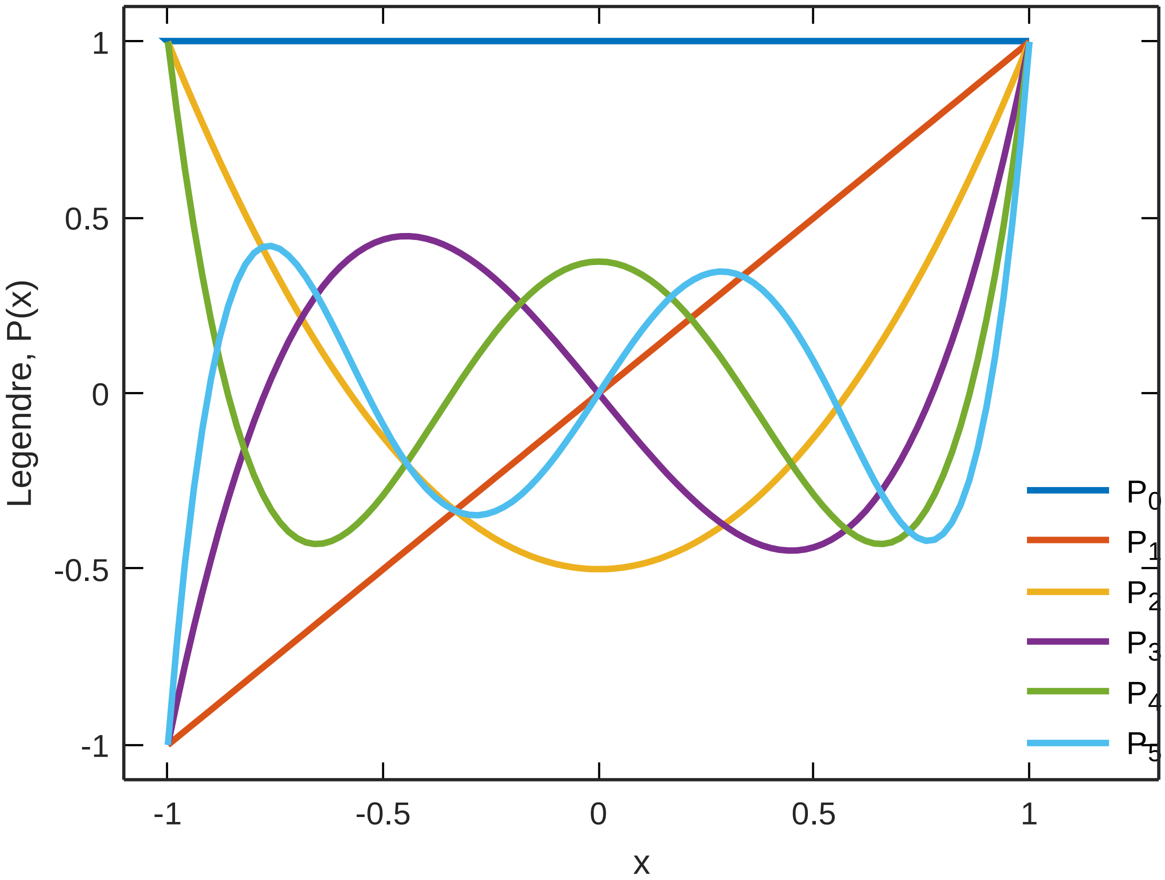

Figure 5. Legendre polynomials

- Forsythe polynomials, $F_k(x)$ are orthogonal over an arbitrary domain $[\alpha, \beta]$ with respect to an arbitrary weight, $w(x)$. Weights used in curve-fitting are typically the magnitude square of the data or the inverse of the measurement error of each data point. The recursive generation of Forsythe polynomials is similar to Graham-Schmidt orthogonalization.

$$F_0(x) = 1 \tag{36}$$

$$F_1(x) = xF_0(x) - C_1F_0(x) = x - C_1 \tag{37}$$

Applying the orthogonality condition to $F_0(x)$ and $F_1(x)$,

$$\int_\alpha^\beta F_0(x)\ w(x)\ F_1(x)\ dx = 0 \tag{38}$$

leads to the condition

$$\int_\alpha^\beta x\ w(x)\ dx = C_1 \int_\alpha^\beta w(x)\ dx. \tag{39}$$

Higher degree polynomials are found by substituting the recurrence relationship

$$F_{k+1}(x) = xF_k(x) - C_{k+1}F_k(x) - D_{k+1}F_{k-1}(x). \tag{40}$$

into the orthogonality condition

$$\int_\alpha^\beta F_{k+1}(x)\ w(x)\ F_k(x)\ dx = 0 \tag{41}$$

and solving for C_{k+1} and D_{k+1} to obtain

$$C_{k+1} = \frac{\int_\alpha^\beta x\ w(x)\ F_k^2(x)\ dx}{\int_\alpha^\beta w(x)\ F_k^2(x)\ dx} \tag{42}$$

$$D_{k+1} = \frac{\int_\alpha^\beta x\ w(x)\ F_k(x)\ F_{k-1}\ dx}{\int_\alpha^\beta w(x)\ F_{k-1}^2(x)\ dx} \tag{43}$$

$$\tag{44}$$

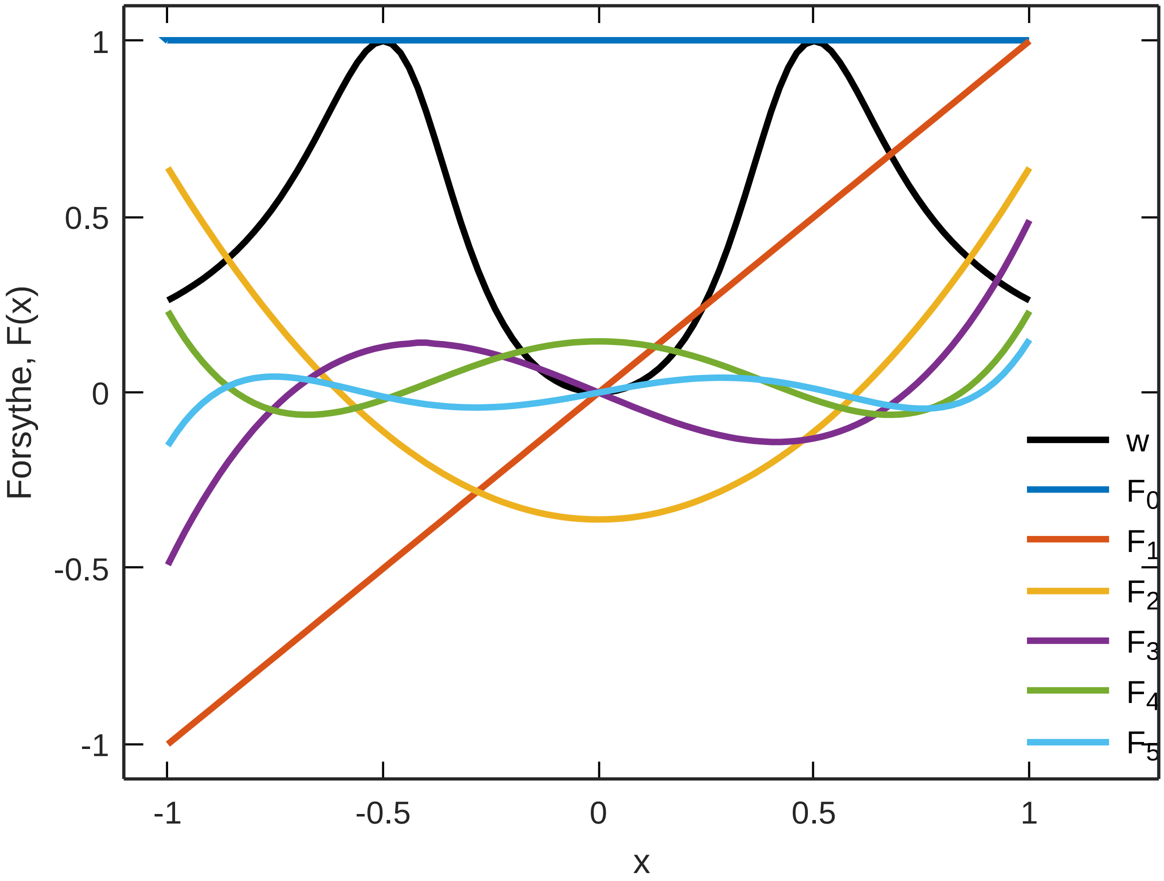

Figure 6. Forsythe polynomials on $[-1,1]$ orthogonal to $w(x) = (0.2x^2)/((x^2-0.25)^2+0.2x^2)$

- Chebyshev polynomials $T_k(x)$ are defined by the trigonometric expression

$$T_k(x) = \cos(k \arccos x), \tag{45}$$

solve of the ordinary differential equation,

$$(1-x^2)T_k''(x) - xT_k'(x) + k^2 T_k(x) = 0 \ ,$$

and are orthogonal with respect to $w(x) = (1-x^2)^{-1/2}$ over the domain $[-1,1]$.

$$\int_{-1}^{1} T_m(x) \, \frac{1}{\sqrt{1-x^2}} \, T_n(x) \, dx = \begin{cases} \pi & m = n = 0 \\ \frac{\pi}{2} & m = n \neq 0 \\ 0 & m \neq n \end{cases} \tag{46}$$

The discrete form of the orthogonality condition for Chebyshev polynomials a special definition because the weighting function for Chebyshev polynomials is not defined at the end-points. Given the P real roots of $T_p(x)$, t_p, $p = 1, \cdots P$, the discrete orthogonality relationship for Chebyshev polynomials is

$$\sum_{p=1}^{P} T_m(t_p) \, T_n(t_p) = \begin{cases} P & m = n = 0 \\ \frac{P}{2} & m = n \neq 0 \\ 0 & m \neq n \end{cases} \tag{47}$$

where

$$t_p = \cos\left(\pi \, \frac{p - 1/2}{P}\right) \quad \text{for} \quad p = 1, \cdots, P$$

Note that the discrete orthogonality relationship, equation (47) is exact, and is not a trapezoidal-rule approximation to the continuous orthogonality relationship, equation (46).

The recurrence relationship for Chebyshev polynomials is simply

$$T_{k+1}(x) = 2xT_k(x) + T_{k-1}(x). \tag{48}$$

Chebyshev polynomials are often associated with an "equi-ripple" or "mini-max" property. If an approximation $\hat{f}(x) \approx \sum_{k=0}^{N} c_k T_k(x)$ has an error $e = y(x) - \hat{y}(x)$ that is dominated by $T_{N+1}(x)$, then the maximum of the approximation error is roughly minimized. This desirable feature indicates that the error is approximately uniform over the domain of the approximation; that the magnitude of the error is no worse in one part of the domain than in another part of the domain.

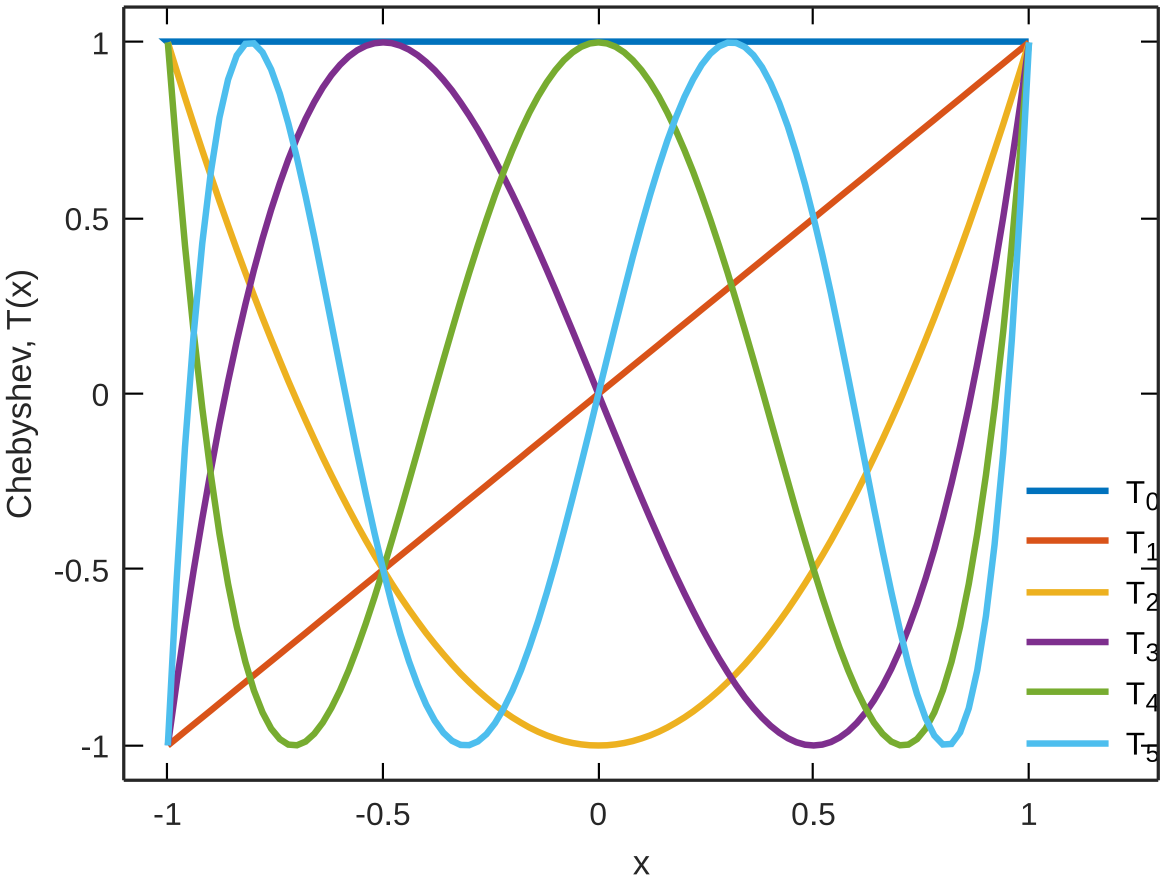

Figure 7. Chebyshev polynomials

8.2 The Application of Orthogonal Polynomials to Curve-Fitting

The benefit of curve-fitting in a basis of orthogonal polynomials is that the normal equations are diagonalized, that the model parameters may be computed directly, without consideration of ill-conditioned systems of equations or numerical linear algebra, and that the parameter errors are uncorrelated.

The cost of curve-fitting in a basis of orthogonal polynomials is that the basis must be constructed in a way that preserves the discrete orthogonality conditions. For Legendre polynomials the values of the independent variables must be uniformly spaced and mapped to the interval $[-1 : 1]$, For Chebyshev polynomials the values of the independent variables must be the roots of a high-order Chebyshev polynomial. In both of these bases the data y must be interpolated to the specified values of the independent variables. For a Forsythe basis, the data need not be mapped or interpolated.

Consider Chebyshev approximation in which the mapping and interpolation steps have already been carried out.

$$e_p = y_p - \hat{y}(t_p; c) = y_p - \sum_{k=0}^{n} c_k T_k(t_p) \tag{49}$$

which may be written for all m data points,

$$\begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} - \begin{bmatrix} 1 & T_1(t_1) & T_2(t_1) & \cdots & T_n(t_1) \\ 1 & T_1(t_2) & T_2(t_2) & \cdots & T_n(t_2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & T_1(t_m) & T_2(t_m) & \cdots & T_n(t_m) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}, \tag{50}$$

or $e = y - Tc$. Values t_p are roots of a high order Chebyshev polynomial and data points y_p have been interpolated to points t_p.

Minimizing the quadratic objective function $J = \sum e_p^2$ leads to the normal equations

$$\hat{c} = [T^{\mathsf{T}}T]^{-1}T^{\mathsf{T}}y, \tag{51}$$

where $[T^{\mathsf{T}}T]$ is diagonal as per the discrete orthogonality relation, equation (47)

$$\begin{aligned}[T^{\mathsf{T}}T] &= \begin{bmatrix} \sum T_1(t_p)T_1(t_p) & \sum T_1(t_p)T_2(t_p) & \sum T_1(t_p)T_3(t_p) & \cdots & \sum T_1(t_p)T_n(t_p) \\ & \sum T_2(t_p)T_2(t_p) & \sum T_2(t_p)T_3(t_p) & \cdots & \sum T_2(t_p)T_n(t_p) \\ & & \sum T_3(t_p)T_3(t_p) & \cdots & \sum T_3(t_p)T_n(t_p) \\ & \text{SYM} & & \ddots & \vdots \\ & & & & \sum T_n(t_p)T_n(t_p) \end{bmatrix} \\ &= \begin{bmatrix} P & 0 & 0 & \cdots & 0 \\ 0 & P/2 & 0 & \cdots & 0 \\ 0 & 0 & P/2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & P/2 \end{bmatrix} \end{aligned} \tag{52}$$

The discrete orthogonality of Chebyshev polynomials is exact to within machine precision, regardless of the number of terms in the summation. Because $[T^{\mathsf{T}}T]$ may be inverted analytically, the curve-fit coefficients may be computed directly from

$$\hat{c}_0 = \frac{1}{P}\sum_{p=1}^{P} y(t_p) \tag{53}$$

$$\hat{c}_k = \frac{2}{P}\sum_{p=1}^{P} T_k(t_p)y(t_p), \qquad \text{for} \qquad k > 0 \tag{54}$$

Also, note that $T_k(t_p) = \cos(k\pi(p - 1/2)/P)$. The cost of the simplicity of the closed-form expression for c_k using the Chebyshev polynomial basis is the need to re-scale the independent variables, x to the interval $[-1, 1]$, and to interpolate the data, y, to the roots of $T_P(x)$.

Coefficients estimated in an orthogonal polynomial basis have a diagonal covariance matrix, as per equations (52) and (16); parameter errors are uncorrelated.

9 Tikhonov Regularization

The goal of regularization is to modify the normal equations $[X^\mathsf{T} X]a = X^\mathsf{T} y$ in order to significantly improve its condition number while leaving the solution a relatively un-changed.

In Tikhonov regularization the least-squares error criterion is augmented with a quadratic term involving the parameters. The effect of Tikhonov regularization is to estimate model parameters while also keeping the model parameters near zero, or near some other set of values. Doing so improves the conditioning of the normal equations.

Consider the over-determined system of linear equations $y = Xa$ where $y \in \mathbb{R}^m$ and $a \in \mathbb{R}^n$, and $m > n$. We seek the solution a that minimizes the quadratic objective function

$$J(a) = ||Xa - y||_P^2 + \beta||a - \bar{a}||_Q^2, \tag{55}$$

where the quadratic vector norm is defined as $||x||_P^2 = x^\mathsf{T} Px$, in which the weighting matrix P is positive definite, and the Tikhonov regularization factor β is non-negative. If the vector y is obtained through imprecise measurements, and the measurements of each element of y_i are statistically independent, then P is typically a diagonal matrix in which each diagonal element P_{ii} is the inverse of the variance of the measurement error of y_i, $P_{ii} = 1/\sigma_{y_i}^2$. If the errors in y_i are not statistically independent, then P should be the inverse of the covariance matrix V_y of the vector y, $P = \mathsf{V}_y^{-1}$. The positive definite matrix Q and the reference parameter vector $\bar{a}$ reflect the way in which we would like to constrain the parameters. For example, we may simply want the solution, a to be near some reference point, $\bar{a}$, in which case $Q = I_n$. Alternatively, we may wish some linear function of the parameters $L_Q a$ to be minimized, in which case $Q = L_Q^\mathsf{T} L_Q$ and $\bar{a} = 0$. Expanding the quadratic objective function,

$$J(a) = a^\mathsf{T} X^\mathsf{T} PXa - 2a^\mathsf{T} X^\mathsf{T} Py + y^\mathsf{T} Py + \beta a^\mathsf{T} Qa - 2\beta a^\mathsf{T} Q\bar{a} + \beta\bar{a}^\mathsf{T} Q\bar{a}. \tag{56}$$

The objective function is minimized by setting the first partial of $J(a)$ with respect to a equal to zero,

$$\frac{\partial J(a)}{\partial a}^\mathsf{T} = 2X^\mathsf{T} PXa - 2X^\mathsf{T} Py + 2\beta Qa - 2\beta Q\bar{a} = 0_{n\times 1}, \tag{57}$$

and solving for the parameter estimates $\hat{a}$,

$$\hat{a}_{(\beta)} = [X^\mathsf{T} PX + \beta Q]^{-1}(X^\mathsf{T} Py + \beta Q\bar{a}). \tag{58}$$

The meaning of the notation $\hat{a}_{(\beta)}$ is that the solution a depends upon the value of the regularization factor, β. The regularization factor weights the relative importance of $||Xa - y||_P^2$ and $||a - \bar{a}||_Q^2$. For problems in which X or $X^\mathsf{T} PX$ are ill-conditioned, small values of β (i.e., small compared to the average of the diagonal elements of $X^\mathsf{T} PX$) can significantly improve the conditioning of the problem.

If the measurement errors of y_i are not individually known, then it is common to set $P = I_n$. Likewise, if the n parameter differences $a - \bar{a}$ are all equally important, then it is customary to set $Q = I_n$. Finally, if we have no set of reference parameters $\bar{a}$, then it is common to set $\bar{a} = 0$. With these simplifications, the solution is given by $\hat{a}_{(\beta)} =$

$[X^\mathsf{T} X + \beta I_n]^{-1} X^\mathsf{T} y$, which may be written $\hat{a}_{(\beta)} = X^+_{(\beta)} y$, where $X^+_{(\beta)}$ is called the regularized *pseudo-inverse*. In the more general case, in which $P \neq I_n$ and $Q \neq I_n$, but $\bar{a} = 0_{n \times 1}$,

$$X^+_{(\beta)} = [X^\mathsf{T} P X + \beta Q]^{-1} X^\mathsf{T} P. \tag{59}$$

The dimension of $X^+_{(\beta)}$ is $n \times m$. In a later section we will see that if L_Q is inevitable then we can always scale and shift X, y, and a with no loss of generality.

9.1 Error Analysis of Tikhonov Regularization

We are interested in determining the covariance matrix of the solution $\hat{a}_{(\beta)}$.

$$\begin{aligned} \mathsf{V}_{\hat{a}(\beta)} &= \left[\frac{\partial \hat{a}_{(\beta)}}{\partial y}\right] \mathsf{V}_y \left[\frac{\partial \hat{a}_{(\beta)}}{\partial y}\right]^\mathsf{T} \\ &= X^+_{(\beta)} \, \mathsf{V}_y \, X^{+\mathsf{T}}_{(\beta)} \\ &= [X^\mathsf{T} P X + \beta Q]^{-1} X^\mathsf{T} P \, \mathsf{V}_y \, P X [X^\mathsf{T} P X + \beta Q]^{-1} \\ &= [X^\mathsf{T} P X + \beta Q]^{-1} X^\mathsf{T} P X [X^\mathsf{T} P X + \beta Q]^{-1}, \end{aligned} \tag{60}$$

where we use $P = \mathsf{V}_y^{-1}$. This covariance matrix is sometimes called the error propagation matrix, as it indicates how random errors in y propagate to the estimates $\hat{a}$. Note that in the special case of no regularization ($\beta = 0$),

$$\mathsf{V}_{\hat{a}(0)} = [X^\mathsf{T} P X]^{-1}, \tag{61}$$

and that the parameter covariance matrix with regularization is always smaller than that without regularization.

In addition to having propagation errors, the estimate $\hat{a}_{(\beta)}$ is biased by the regularization factor. Let us presume that we know y exactly, and that the exact value of y is y_e. The exact solution, without regularization, is $a_\mathrm{e} = [X^\mathsf{T} P X]^{-1} X^\mathsf{T} P y_\mathrm{e}$. The regularization error, $\delta a_{(\beta)} = \hat{a}_{(\beta)} - a_\mathrm{e}$ (for $\bar{a} = 0$), is

$$\begin{aligned} \delta a_{(\beta)} &= [X^+_{(\beta)} - X^+_{(0)}] y_\mathrm{e} \\ &= [[X^\mathsf{T} P X + \beta Q]^{-1} X^\mathsf{T} P - [X^\mathsf{T} P X]^{-1} X^\mathsf{T} P] y_\mathrm{e} \\ &= [[X^\mathsf{T} P X + \beta Q]^{-1} - [X^\mathsf{T} P X]^{-1}] X^\mathsf{T} P y_\mathrm{e}. \end{aligned} \tag{62}$$

Recall that $a_\mathrm{e} = [X^\mathsf{T} P X]^{-1} X^\mathsf{T} P y_\mathrm{e}$, or $X^\mathsf{T} P X a_\mathrm{e} = X^\mathsf{T} P y_\mathrm{e}$, so

$$\begin{aligned} \delta a_{(\beta)} &= [[X^\mathsf{T} P X + \beta Q]^{-1} - [X^\mathsf{T} P X]^{-1}] X^\mathsf{T} P X a_\mathrm{e} \\ &= [[X^\mathsf{T} P X + \beta Q]^{-1} X^\mathsf{T} P X - I_n] a_\mathrm{e} \\ &= [X^\mathsf{T} P X + \beta Q]^{-1} [X^\mathsf{T} P X + \beta Q - \beta Q] a_\mathrm{e} - I_n a_\mathrm{e} \\ &= [X^\mathsf{T} P X + \beta Q]^{-1} [X^\mathsf{T} P X + \beta Q] a_\mathrm{e} - [X^\mathsf{T} P X + \beta Q]^{-1} \beta Q a_\mathrm{e} - I_n a_\mathrm{e} \\ &= -[X^\mathsf{T} P X + \beta Q]^{-1} Q \beta a_\mathrm{e}. \end{aligned} \tag{63}$$

The regularization error $\delta a_{(\beta)}$ equals zero if $\beta = 0$ and increases with β.

The total mean squared error matrix, $E_{(\beta)}$ is the sum of the parameter covariance matrix and the regularization bias error, $E_{(\beta)} = \mathsf{V}_{\hat{a}(\beta)} + \delta a_{(\beta)} \delta a^\mathsf{T}_{(\beta)}$. The mean squared error is the trace of $E_{(\beta)}$.

10 Singular Value Decomposition

Consider a real matrix X that is not necessarily square, $X \in \mathbb{R}^{m\times n}$, with $m > n$. The rank of X is r and $r \leq n$. Let $\lambda_1, \lambda_2, \cdots, \lambda_r$ be the positive eigenvalues of $[X^\mathsf{T} X]$, including multiplicity, ordered in decreasing numerical order, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_r > 0$. The *singular values*, σ_i of X are defined as the square roots of the eigenvalues, $\sigma_i = \sqrt{\lambda_i}$, $i = 1, \cdots, r$.

The singular value decomposition of a matrix X ($X \in \mathbb{R}^{m\times n}$, $m \geq n$) is given by the factorization

$$X = U\Sigma V^\mathsf{T} \tag{64}$$

where $U \in \mathbb{R}^{m\times m}$, $\Sigma \in \mathbb{R}^{m\times n}$ and $V \in \mathbb{R}^{n\times n}$. The matrices U and V are orthonormal, $U^\mathsf{T} U = I_m$ and $V^\mathsf{T} V = I_n$, and Σ is a diagonal matrix of the singular values of X, $\Sigma = \mathrm{diag}(\sigma_1\ \sigma_2\ \cdots\ \sigma_n)$. The singular values of X are sorted in a non-increasing numerical order, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. The number of singular values equal to zero is equal to the number of linearly dependent columns of X. If r is the rank of X then $n - r$ singular values are equal to zero. The ratio of the maximum to the minimum singular value is called the *condition number* of X, $c_X = \sigma_1/\sigma_n$. Matrices with very large condition numbers are said to be *ill-conditioned.* If $\sigma_n = 0$, then $c_X = \infty$ and X is said to be *singular*, and is non-invertible.

The columns of U and V are called the right and left *singular vectors*, $U = [u_1\, u_2\ \cdots\ u_m]$ and $V = [v_1\ v_2\ \cdots\ v_n]$. The left singular vectors u_i are column vectors of dimension m and the right singular vectors v_i are column vectors of dimension n. If one or more singular value of X is equal to zero, $r < n$, and the set of right singular vectors $\{v_{r+1} \cdots v_n\}$ (corresponding to $\sigma_{r+1} = \cdots = \sigma_n = 0$) form an orthonormal basis for the *null-space* of X. The dimension of the null space of X plus the rank of X equals n.

The singular value decomposition of X may be written as an expansion of the singular vectors,

$$X = \sum_{i=1}^{n} \sigma_i [u_i v_i^\mathsf{T}], \tag{65}$$

where the rank-1 matrices $[u_i v_i^\mathsf{T}]$ have the same dimension as X. Also note that $||u_i||_I = ||v_i||_I = 1$. Therefore, the significance of each term of the expansion $\sigma_i u_i v_i^\mathsf{T}$ decreases with i.

The system of equations $y = Xa$ may be inverted using singular value decomposition: $a = V\Sigma^{-1}U^\mathsf{T} y$, or

$$a = \sum_{i=1}^{n} \frac{1}{\sigma_i} v_i u_i^\mathsf{T} y. \tag{66}$$

The singular values in the expansion which contribute least to the decomposition of X can potentially dominate the solution a. An additive perturbation δy in y will propagate to a perturbation in the solution, $\delta a = V\Sigma^{-1}U^\mathsf{T}\delta y$. The magnitude of δa in the direction of v_i is equal to the dot product of u_i with δy divided by σ_i,

$$v_i^\mathsf{T}\delta a = \frac{1}{\sigma_i} u_i^\mathsf{T}\delta y. \tag{67}$$

Therefore, perturbations δy that are orthogonal to all of the left singular vectors, u_i, are not propagated to the solution. Conversely, any perturbation δa in the direction of v_i contributes to y in the direction of u_i by an amount equal to $\sigma_i||\delta a||$.

If the rank of X is less than n, $(r < n)$, the components of a which lie in the space spanned by $\{v_{r+1} \ \cdots \ v_n\}$ will have no contribution to y. In principle, this means that any component of y in the space spanned by the sub-set of left singular vectors $\{u_{r+1} \ \cdots \ u_m\}$ is an "error" or is "noise", since it can not be obtained using the expression $y = Xa$ for any value of a. These components of y are called "noise" or "error" because they can not be predicted by the model equations, $y = Xa$. In addition, these "noisy" components of y, which lie in the space $\{u_{r+1} \ \cdots \ u_m\}$, will be magnified to an infinite degree when used to identify the model parameters a.

From the singular value decomposition of X we find that $X^\mathsf{T} X = V \Sigma^2 V^\mathsf{T}$. If c is the condition number of X, then the condition number of $X^\mathsf{T} X$ is c^2. If c is large, solving $[X^\mathsf{T} X]a = X^\mathsf{T} y$ can be numerically treacherous.

When X is obtained using measured data, it is almost never singular but is often ill-conditioned. We will consider two types of ill-conditioned matrices: (*i*) matrices in which the first r singular values are all much larger than the last $n-r$ singular values, and (*ii*) matrices in which the singular values decrease at a rate that is more or less uniform.

10.1 Truncated Singular Value Expansion

If the first r singular values are much larger than the last $n-r$ singular values, (i.e., $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r >> \sigma_{r+1} \geq \sigma_{r+2} \geq \cdots \geq \sigma_n \geq 0$), then a relatively accurate representation of X may be obtained by simply retaining the first r singular values of X and the corresponding columns of U and V,

$$X_{(r)} = U_r \Sigma_r V_r^\mathsf{T} = \sum_{i=1}^{r} \sigma_i u_i v_i^\mathsf{T} , \tag{68}$$

and the truncated expression for the parameter estimates is

$$\hat{a}_{(r)} = V_r \Sigma_r^{-1} V_r^\mathsf{T} y = \sum_{i=1}^{r} \frac{1}{\sigma_i} v_i u_i^\mathsf{T} y \ , \tag{69}$$

where V_r and U_r contain the first r rows of V and U, and where Σ_r contains the first r rows and columns of Σ.

If σ_{r+1} is close to the numerical precision of the computation, ϵ ($\epsilon \approx 10^{-6}$ for single precision and $\epsilon \approx 10^{-12}$ for double precision), then the singular values, $\sigma_{r+1} \ \cdots \ \sigma_n$ and the corresponding columns of U and V contribute negligibly to X. Their contribution to the solution vector, a, can be dominated by random noise and round-off error in y.

The parameter estimate covariance matrix is derived using equation (4), in which

$$\left[\frac{\partial \hat{a}}{\partial y} \right] = V_r \Sigma_r^{-1} U_r^\mathsf{T} , \tag{70}$$

Therefore the covariance matrix of the parameter estimates is

$$\mathsf{V}_{\hat{a}(r)} = V_r \Sigma_r^{-1} U_r^\mathsf{T} \, \mathsf{V}_y \, U_r \Sigma_r^{-1} V_r^\mathsf{T} . \tag{71}$$

Because the solution $\hat{a}_{(r)}$ does not contain components that are close to the null-space of X the covariance matrix is limited to the range of X for which the singular values are not unacceptably large. The cost of the reduced parameter covariance matrix is an increased bias error, introduced through truncation. Assuming that we know y exactly, the corresponding exact solution with no regularization a_{e} can be used to evaluate regularization bias error, $\delta a_{(r)} = \hat{a}_{(r)} - a_{\mathrm{e}}$.

$$\delta a_{(r)} = V_r \Sigma_r^{-1} U_r^{\mathsf{T}} y_{\mathrm{e}} - V \Sigma^{-1} U^{\mathsf{T}} y_{\mathrm{e}} \tag{72}$$

Substituting $V \Sigma U^{\mathsf{T}} = U_r \Sigma_r V_r^{\mathsf{T}} + U_n \Sigma_n V_n^{\mathsf{T}}$,

$$\delta a_{(r)} = -[V_n \Sigma_n^{-1} U_n^{\mathsf{T}}] y_{\mathrm{e}}, \tag{73}$$

where V_n and U_n contain the last $n-r$ rows of V and U, and where Σ_n contains the last $n-r$ rows and columns of Σ. Substituting $y_{\mathrm{e}} = U \Sigma V^{\mathsf{T}} a_{\mathrm{e}}$,

$$\delta a_{(r)} = -[V_n \Sigma_n^{-1} U_n^{\mathsf{T}} U \Sigma V^{\mathsf{T}}] a_{\mathrm{e}} \tag{74}$$

Noting that $U_n^{\mathsf{T}} U = \Sigma_n^{-1} U_n^{\mathsf{T}} U \Sigma = [0_{n-r \times r} \ I_{n-r}]$,

$$\delta a_{(r)} = -[V_n V_n^{\mathsf{T}}] a_{\mathrm{e}} \tag{75}$$

Note that while the matrix $-V_n^{\mathsf{T}} V_n$ equals I_{n-r}, the matrix $-V_n V_n^{\mathsf{T}}$ is not identity because the summation is only over the last $n-r$ rows of V.

The total mean squared error matrix, $E_{(r)}$ is the sum of the parameter covariance matrix and the truncation bias error, $E_{(r)} = V_r \Sigma_r^{-1} U_r^{\mathsf{T}} \mathsf{V}_y \, U_r \Sigma_r^{-1} V_r^{\mathsf{T}} - V_n V_n^{\mathsf{T}} a_{\mathrm{e}}$. The mean squared error is the trace of $E_{(r)}$.

10.2 Singular Value Decomposition with Tikhonov Regularization

If the singular values decrease at a rate that is more or less uniform, then selecting r for the truncated approximations above may require some subjective reasoning. Certainly any singular value that is equal to or less than the precision of the computation should be eliminated. However, if $\sigma_1 \approx 10^{15}$ and $\sigma_n \approx 10^{-3}$, X would be considered ill conditioned by most standards, even though the smallest singular value can be resolved even with single-precision computations. As an alternative to eliminating one or more of the smallest singular values one may simply add a small constant β to all of the singular values. This can substantially improve the condition number of the system without eliminating any of the information contained in the full singular value factorization. This approach is equivalent to Tikhonov regularization.

To link the formulation of Tikhonov regularization to singular value decomposition, it is useful to show that a the Tikhonov objective function (55) may be written

$$J(\tilde{a}) = ||\tilde{X}\tilde{a} - \tilde{y}||_I^2 + \beta ||\tilde{a}||_I^2 + C, \tag{76}$$

by simply scaling and shifting X, y, and a, and with no loss of generality. In the above expression, C is independent of $\tilde{a}$ and does not affect the parameter estimates. Defining L_P

and L_Q as the Cholesky factors of P and Q, and defining $\tilde{X} = L_P X L_Q^{-1}$, $\tilde{y} = L_P(y - X\bar{a})$, and $\tilde{a} = L_Q(a - \bar{a})$ then equation (55) is equivalent to equation (76), where

$$C = \bar{a}^{\mathsf{T}} X^{\mathsf{T}} L_P^{\mathsf{T}} L_P X \bar{a} - 2\bar{a}^{\mathsf{T}} X^{\mathsf{T}} L_P^{\mathsf{T}} L_P y \tag{77}$$

Setting $[\partial J(\tilde{a})/\partial \tilde{a}]^{\mathsf{T}}$ to zero results in the least-squares parameter estimates

$$\hat{\tilde{a}}_{(\beta)} = [\tilde{X}^{\mathsf{T}} \tilde{X} + \beta I]^{-1} \tilde{X}^{\mathsf{T}} \tilde{y}. \tag{78}$$

Note that the solutions $\hat{a}_{(\beta)}$ and $\hat{\tilde{a}}_{(\beta)}$ are related by the scaling

$$\hat{a}_{(\beta)} = L_Q^{-1} \hat{\tilde{a}}_{(\beta)} + \bar{a}. \tag{79}$$

In other words, the minimum value of the objective function (55) coincides with the minimum value of the objective function (76). As a simple example of the effects of scaling on the parameter estimates, consider the two equivalent quadratic objective functions $J(a) = 5a^2 - 3a + 1$ and $J(\tilde{a}) = 20\tilde{a}^2 - 6\tilde{a} + 2$, where a is scaled, $a = 2\tilde{a}$. The parameter estimates $\hat{a} = 3/10$ and $\hat{\tilde{a}} = 3/20$. These estimates satisfy the scaling relationship, $\hat{a} = 2\hat{\tilde{a}}$.

The singular value decomposition of $\tilde{X}$ may be substituted into the least-squares solution for $\hat{\tilde{a}}_{(\beta)}$

$$\begin{aligned}
\hat{\tilde{a}}_{(\beta)} &= [\tilde{V}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{U}\tilde{\Sigma}\tilde{V}^{\mathsf{T}} + \beta I]^{-1} \tilde{V}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{y} \\
&= [\tilde{V}\tilde{\Sigma}^2\tilde{V}^{\mathsf{T}} + \beta \tilde{V} I \tilde{V}^{\mathsf{T}}]^{-1} \tilde{V}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{y} \\
&= [\tilde{V}(\tilde{\Sigma}^2 + \beta I)\tilde{V}^{\mathsf{T}}]^{-1} \tilde{V}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{y} \\
&= \tilde{V}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{V}^{\mathsf{T}}\tilde{V}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{y} \\
&= \tilde{V}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{\Sigma}\tilde{U}^{\mathsf{T}}\tilde{y}
\end{aligned} \tag{80}$$

The covariance of the parameter errors is largest in the direction corresponding to the maximum value of $\tilde{\sigma}_i/(\tilde{\sigma}_i^2 + \beta)$. If $\tilde{X}$ is singular, then as β approaches zero, random errors propagate in a direction which is close to the null space of $\tilde{X}$. Note that the singular value decomposition solution to $\tilde{y} = \tilde{X}\tilde{a}$ is $\tilde{a} = \tilde{V}\tilde{\Sigma}^{-1}\tilde{U}^{\mathsf{T}}\tilde{y}$. Thus, Tikhonov regularization is equivalent to a singular value decomposition solution, in which the inverse of each singular value, $1/\tilde{\sigma}_i$, is replaced by $\tilde{\sigma}_i/(\tilde{\sigma}_i^2 + \beta)$, or in which each singular value $\tilde{\sigma}_i$ is replaced by $\tilde{\sigma}_i + \beta/\tilde{\sigma}_i$. Thus, the largest singular values are negligibly affected by regularization, while the effects of the smallest singular values on the solution are suppressed, as shown in Figure 8.

10.3 Error Analysis of Singular Value Decomposition with Tikhonov Regularization

The parameter covariance matrix is derived using equation (4), in which

$$\frac{\partial \hat{\tilde{a}}}{\partial \tilde{y}} = \tilde{V}\tilde{\Sigma}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{U}^{\mathsf{T}} \tag{81}$$

and

$$\mathsf{V}_{\tilde{y}} = L_P \, \mathsf{V}_y \, L_P^{\mathsf{T}}. \tag{82}$$

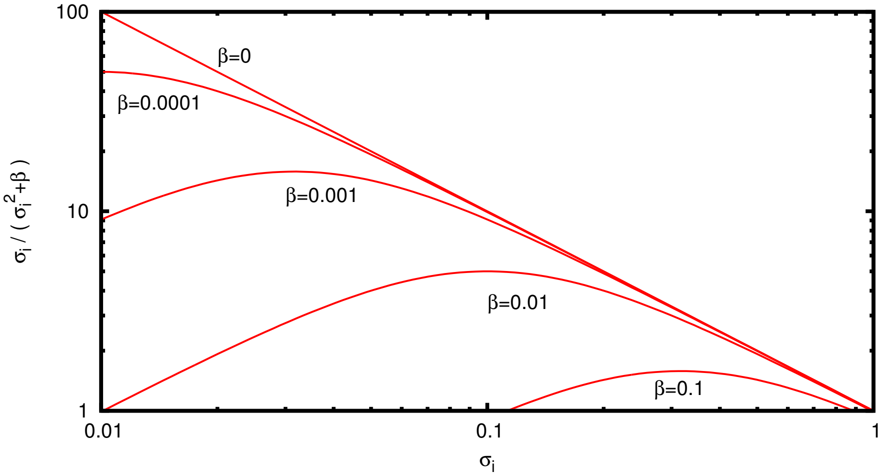

Figure 8. Effect of regularization on singular values.

Recall that V_y is the covariance matrix of the measurement errors, and that $P = \mathsf{V}_y^{-1} = L_P^{\mathsf{T}} L_P$. Using the fact that $(AB)^{-1} = B^{-1}A^{-1}$, we find that $\mathsf{V}_{\tilde{y}} = I$. Therefore the covariance matrix of the solution is

$$\mathsf{V}_{\hat{\tilde{a}}(\beta)} = \tilde{V}\tilde{\Sigma}^2(\tilde{\Sigma}^2 + \beta I)^{-2}\tilde{V}^{\mathsf{T}}. \tag{83}$$

Because $[\partial a/\partial \tilde{a}] = L_Q^{-1}$, the covariance matrix of the original solution is

$$\mathsf{V}_{\hat{a}(\beta)} = L_Q^{-1}\tilde{V}\tilde{\Sigma}^2(\tilde{\Sigma}^2 + \beta I)^{-2}\tilde{V}^{\mathsf{T}} L_Q^{-T}. \tag{84}$$

As in equation (60), we see here that increasing β reduces the covariance of the propagated error quadratically. The cost of the reduced parameter covariance matrix is a bias error, introduced through regularization. Assuming that we know $\tilde{y}$ exactly, the corresponding exact solution with no regularization $\tilde{a}_{\mathrm{e}}$ can be used to evaluate regularization bias error, $\delta\tilde{a}_{(\beta)} = \hat{\tilde{a}}_{(\beta)} - \tilde{a}_{\mathrm{e}}$.

$$\begin{aligned} \delta\tilde{a}_{(\beta)} &= \tilde{V}\tilde{\Sigma}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{U}^{\mathsf{T}}\tilde{y}_{\mathrm{e}} - \tilde{V}\tilde{\Sigma}^{-1}\tilde{U}^{\mathsf{T}}\tilde{y}_{\mathrm{e}} \\ &= [\tilde{V}\tilde{\Sigma}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{U}^{\mathsf{T}} - \tilde{V}\tilde{\Sigma}^{-1}\tilde{U}^{\mathsf{T}}]\tilde{y}_{\mathrm{e}} \end{aligned} \tag{85}$$

Substituting $y_{\mathrm{e}} = \tilde{U}\tilde{\Sigma}\tilde{V}^{\mathsf{T}} a_{\mathrm{e}}$, and $\tilde{U}^{\mathsf{T}}\tilde{U} = I$

$$\begin{aligned} \delta\tilde{a}_{(\beta)} &= [\tilde{V}(\tilde{\Sigma}^2 + \beta I)^{-1}\tilde{\Sigma}^2\tilde{V}^{\mathsf{T}} - I]\tilde{a}_{\mathrm{e}} \\ &= [\tilde{V}(\tilde{\Sigma}^2 + \beta I)^{-1}(\tilde{\Sigma}^2 + \beta I - \beta I)\tilde{V}^{\mathsf{T}} - I]\tilde{a}_{\mathrm{e}} \\ &= -\tilde{V}(\tilde{\Sigma}^2 + \beta I)^{-1}\beta\tilde{V}^{\mathsf{T}}\tilde{a}_{\mathrm{e}} \end{aligned} \tag{86}$$

As in equation (63), we see here that bias errors due to regularization increase with β. In fact, the singular values participating in the bias errors increase β increases. If $\tilde{X}$ is singular, then the exact parameters a_{e} can not lie in the null space of $\tilde{X}$ and the bias error $\delta\tilde{a}_{(\beta)}$ will be orthogonal to the null space of $\tilde{X}$.

The total mean squared error matrix, $\tilde{E}_{(\beta)}$ is the sum of the scaled parameter covariance matrix and the regularization bias error, $\tilde{E}_{(\beta)} = \mathsf{V}_{\tilde{a}(\beta)} + \delta\tilde{a}_{(\beta)}\delta\tilde{a}_{(\beta)}^{\mathsf{T}}$. The mean squared error is the trace of $\tilde{E}_{(\beta)}$.

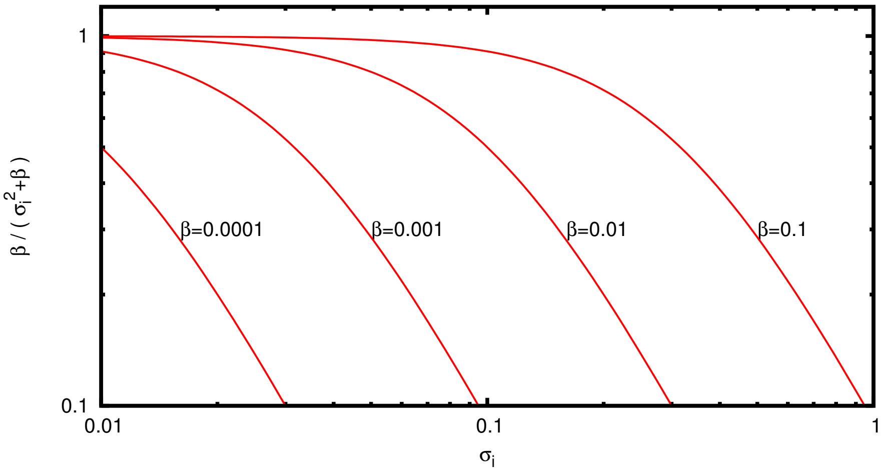

Figure 9. Effect of regularization on bias errors.

10.4 Scaling of Units and Singular Value Decomposition

In most instances, elements of y and a have dissimilar units. In such cases the conditioning of X is affected by the system of units used to describe the elements of y and a (and X). The measurements y and the parameters a may be scaled using diagonal matrices D_y and D_a, $y = D_y \tilde{y}$, $a = D_a \tilde{a}$. Since $D_y \tilde{y} = X D_a \tilde{a}$, $\tilde{y} = D_y^{-1} X D_a \tilde{a}$, and the system matrix for the scaled system is $\tilde{X} = D_y^{-1} X D_a$. Defining SVD's: $X = U \Sigma V^{\mathsf{T}}$ and $\tilde{X} = \tilde{U} \tilde{\Sigma} \tilde{V}^{\mathsf{T}}$, the effect of scaling on the singular values is apparent $\tilde{\Sigma} = \tilde{U}^{\mathsf{T}} D_y^{-1} U \Sigma V^{\mathsf{T}} D_a \tilde{V}$.

Parameter scaling matrices can be designed to achieve a desired spectrum of singular values. But this appears to be a nonlinear problem requiring an iterative method to update D_a and D_y such that the condition number of $D_y^{-1} X D_a$ converges to a minimum value, while possibly meeting other constraints, such as bounds on the values of D_a and D_y.

11 Numerical Example

Consider the singular system of equations $y_o = X_o\ a$,

$$\begin{bmatrix} 100 \\ 1000 \end{bmatrix} = \begin{bmatrix} 1 & 10 \\ 10 & 100 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} \tag{87}$$

The singular value decomposition of X_o is

$$\begin{aligned} U &= \begin{bmatrix} -0.0995037 & -0.995037 \\ -0.995037 & 0.0995037 \end{bmatrix} \\ \Sigma &= \begin{bmatrix} 101 & 0 \\ 0 & 0 \end{bmatrix} \\ V &= \begin{bmatrix} -0.0995037 & 0.995037 \\ -0.995037 & -0.0995037 \end{bmatrix} \end{aligned}$$

Even though X_o is a singular matrix, the estimates for this particular system is defined. The second column of V gives the one-dimensional null-space of X_o. Note that y_o in equation (87) is normal to u_2. Vectors y_o normal to the space spanned by U_n are "noise-free" in the sense that no components of y_o propagates to the null space of X_o. The singular value expansion for the estimates gives

$$\begin{aligned} \begin{bmatrix} \hat{a}_1 \\ \hat{a}_2 \end{bmatrix} &= \frac{1}{101} \begin{bmatrix} -0.0995037 \\ -0.995037 \end{bmatrix} \begin{bmatrix} -0.0995037 & -0.995037 \end{bmatrix} \begin{bmatrix} 100 \\ 1000 \end{bmatrix} + \\ & \quad \frac{1}{0} \begin{bmatrix} 0.995037 \\ -0.0995037 \end{bmatrix} \begin{bmatrix} -0.995037 & 0.0995037 \end{bmatrix} \begin{bmatrix} 100 \\ 1000 \end{bmatrix} + \\ &= \begin{bmatrix} 0.99009900 \\ 9.9009900 \end{bmatrix} \end{aligned} \tag{88}$$

The zero singular value does not affect the estimates because y is orthogonal to u_2. It is interesting to note that despite the fact that the two equations in equation (87) represent the same line (infinitely many solutions) the SVD provides a unique solution, (provided that 0/0 is evaluated to 0).

Regularization works very well for problems in which $U_n^{\mathsf{T}} y$ is very small. Applying regularization to this problem we seek estimates $\hat{a}$ that minimizes the quadratic objective function of equation (55). In other words, we want to find the solution to $y = X_o a$, while keeping the estimates, $\hat{a}$ close to $\bar{a}$. How much we care that $\hat{a}$ is close to $\bar{a}$ is determined by the regularization parameter, β. In general β should be some small fraction of the average of the diagonal elements of X, $\beta << \mathsf{trace}(X)/n$. Increasing β will make the problem easier to solve numerically, but will also add bias to the estimates. The philosophy of using the regularization parameter is something like this: Let's say we have a problem which we can't solve, (i.e., $\det(X) = 0$). Regularization slightly changes the problem into a problem that does have a solution which is ideally independent of the amount of the perturbation.

Because the estimate $\hat{a}$ depends upon β, we can plot a_1 and a_2 vs. β and determine the effect of β on the parameter estimates.

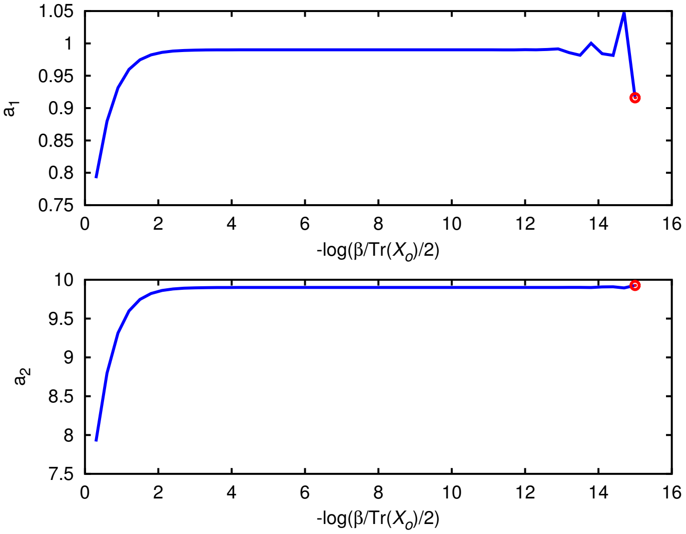

Figure 10. The effect of regularization on solutions to $y_o = [X_o + \beta I]a$ ... where $U_n^{\mathsf{T}} y_o = 0$

From figure 10 we see that as β approaches 0, the estimate approaches $\hat{a}_{(\beta)} = [\ 0.99010\ ;\ 9.90099\ ]$. For this data, the estimate is insensitive to β for $10^{-3} > \beta/\mathsf{trace}(X_o)/2 > 10^{-12}$; for $\beta < 10^{-12}\mathsf{trace}(X_o)/2$ the solution can not be found. Note that for $y = X_o\ a$ as defined above, there are infinitely many solutions; the two lines $a_1(a_2)$ over-lay one another. A solution exists, but it is not unique. In this case a very small regularization factor, β, gives a unique solution. Changing the problem only slightly, by setting $y = [100\ \ 1001]^{\mathsf{T}}$, we see that *no* solution exists to the original problem $y = X_o\ a$. By changing one element of y by only 0.1 percent, the original problem changes from having infinite solutions to having no solution. For systems with no solution, the regularized solution is very sensitive to the choice of the regularization factor, β. There is a region, $10^{-3} > \beta/\mathsf{trace}(X_o)/2 > 10^{-4}$ in this problem, for which a_1 is relatively insensitive to β, however there is no region in which $\frac{da_2}{d\beta} = 0$. For this type of problem, regularization of some type is necessary to find *any* solution, and the solution is sensitive to the value chosen for the regularization parameter.

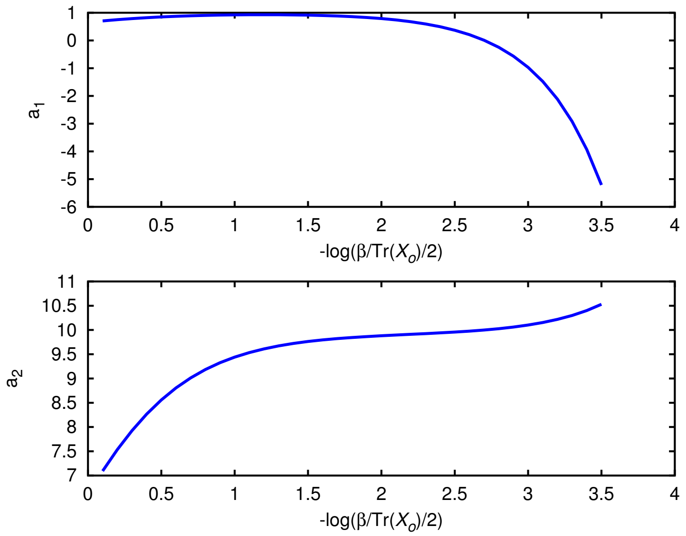

Figure 11. The effect of regularization on parameter estimates to $y + \delta y = [X_o + \beta I]a$ ... where $U_n^{\mathsf{T}}(y_o + \delta y) \neq 0$

Now let's examine the effects of some small random perturbations in both X_o and y. In this part of the example, small random numbers (normally distributed with a mean of zero and a standard deviation of 0.0005) are added to X_o and y, and regularized solutions are found for $\beta = 0.01\mathsf{trace}(X_o)/2$ and $\beta = 0.0001\mathsf{trace}(X_o)/2$, i.e, the equations $y + \delta y = [X_o + \delta X + \beta I]a$ are solved for a. Regularized solutions to these randomly perturbed problems illustrate that

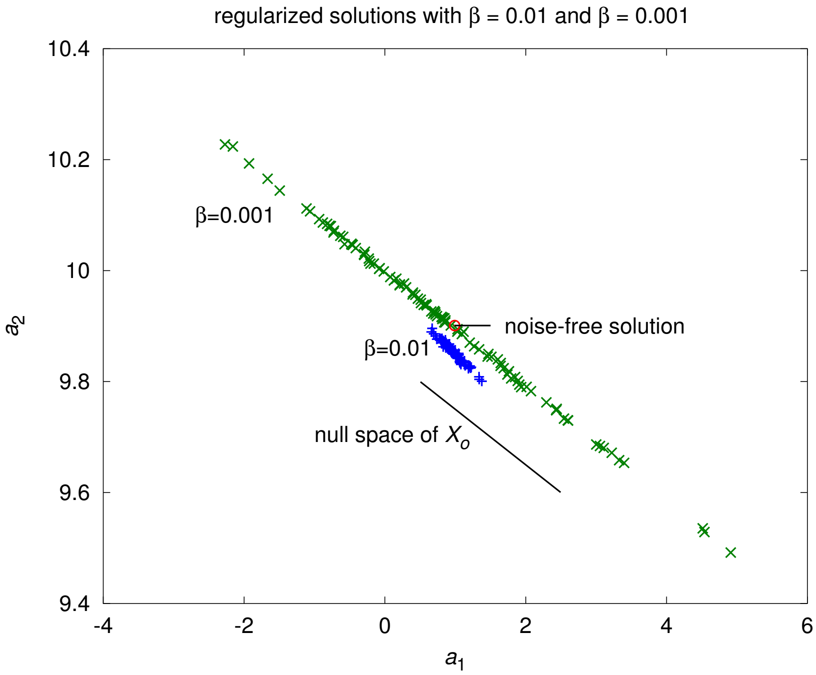

Figure 12. Effect of regularization on the solution of $y + \delta y = [X_o + \delta X + \beta I]a$

at a regularization factor, $\beta \approx 0.01\mathsf{trace}(X_o)/2$, the solution is relatively insensitive to the value of β. Comparing the solutions for 100 randomly perturbed problems, we find that the variance among the solutions is less than 1. For a regularization factor of 0.001, on the other hand, we see that the variance of the solution is quite a bit larger, but that the mean value of all of the solutions is much closer to the noise-free solution. This illustrates the fact that larger values of β decrease the propagation error but introduce a bias error. If no regularization is used, then a_1 and a_2 range from -1000 to +1000 and from -100 to 100, respectively, in this Monte Carlo analysis. Adding the small levels of noise to the non-invertible matrix X_o makes it invertible, however, the solution depends largely on the noise level. In fact if no regularization is used, the solution to this problem depends almost entirely on the noise in the matrices. Increasing the regularization factor, β, reduces the propagation of random noise in the solution at the cost of a bias error.

12 Constrained Least Squares

Suppose that in addition to minimizing the sum-of-squares-of-errors, the curve-fit must also satisfy other criteria. For example, suppose that the curve-fit must pass through a particular point (x_c, y_c), or that the slope of the curve at a particular location, x_s, must be exactly a given value, y'_s. Equality constraints such as these are linear in the parameters and are a natural application of the method of Lagrange multipliers. In general, equality constraints that are linear in the parameter may be expressed as $Ca = b$. The constrained least-squares problem is to minimize $\chi^2(a)$ such that $Ca = b$. The augmented objective function (the *Lagrangian*) becomes,

$$\chi^2_A(a,\lambda) = a^{\mathsf{T}} X^{\mathsf{T}} \mathsf{V}_y^{-1} X a - a^{\mathsf{T}} X^{\mathsf{T}} \mathsf{V}_y^{-1} y - y^{\mathsf{T}} \mathsf{V}_y^{-1} X a + y^{\mathsf{T}} \mathsf{V}_y^{-1} y + \lambda^{\mathsf{T}}(Ca - b) \qquad (89)$$

Minimizing χ^2_A with respect to a and maximizing χ^2_A with respect to λ results in a system of linear equations for the coefficient estimates $\hat{a}$ and Lagrange multipliers $\hat{\lambda}$.

$$\begin{bmatrix} 2X^{\mathsf{T}} \mathsf{V}_y^{-1} X & C^{\mathsf{T}} \\ C & 0 \end{bmatrix} \begin{bmatrix} \hat{a} \\ \hat{\lambda} \end{bmatrix} = \begin{bmatrix} 2X^{\mathsf{T}} \mathsf{V}_y^{-1} y \\ b \end{bmatrix} \qquad (90)$$

If the curve-fit problem has n coefficients and c constraint equations, then the matrix is square and of size $(n + c) \times (n + c)$.

13 Recursive Least Squares

When data points are provided sequentially, parameter estimates $a_{(m)}$ can be updated with each new observation, (x_{m+1}, y_{m+1}).

Given a set of m measurement points and estimates of model parameters $a_{(m)}$ corresponding to the data set (x_i, y_i), $i = 1, ..., m$, we seek an update to the model parameters $a_{(m+1)}$ from a new measurement x_{m+1}, y_{m+1}.

Presuming a linear model, $\hat{y}(x; a) = Xa$, we define the i-th rows of X as $\bar{x}_i$ so that

$$\begin{bmatrix} \hat{y}_1 \\ \vdots \\ \hat{y}_m \end{bmatrix} = \begin{bmatrix} - & \bar{x}_1 & - \\ & \vdots & \\ - & \bar{x}_m & - \end{bmatrix} \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$$

The least-squares error criterion is

$$\begin{aligned} J_{(m)} &= \sum_{i=1}^{m} ||y_i - \hat{y}_i||_F^2 \\ &= \sum_{i=1}^{m} (y_i - \bar{x}_i a)^{\mathsf{T}} (y_i - \bar{x}_i a) \\ &= \sum_{i=1}^{m} (y_i^{\mathsf{T}} y_i - 2 y_i^{\mathsf{T}} \bar{x}_i a + a^{\mathsf{T}} \bar{x}_i^{\mathsf{T}} \bar{x}_i a) \end{aligned}$$

and applying the necessary condition for optimality,

$$\left(\frac{\partial J_{(m)}}{\partial a} \right)^{\mathsf{T}} = 0 \ : \ \sum_{i=1}^{m} [\bar{x}_i^{\mathsf{T}} \bar{x}_i] \hat{a}_{(m)} = \sum_{i=1}^{m} \bar{x}_i^{\mathsf{T}} y_i$$

Defining some new terms to simplify the notation,

$$R_{(m)} = \sum_{i=1}^{m} [\bar{x}_i^{\mathsf{T}} \bar{x}_i]$$

and

$$q_{(m)} = \sum_{i=1}^{m} \bar{x}_i^{\mathsf{T}} y_i \ ,$$

we see that $R_{(m)}$ is a sum of rank-1 matrices. and it can be viewed as an auto-correlation of the sequence of the rows $\bar{x}_i$ of the model basis X, in the special case that x_i have a mean of zero. The matrix $R_{(m)}^{-1}$ is interpreted as the parameter estimate covariance matrix, $[X^{\mathsf{T}} X]^{-1}$. The (column) vector $q_{(m)}$ can be viewed as a cross-correlation between the rows of the model basis and the data, in the special case that x_i and y_i have a mean of zero. Given a new measurement (x_{m+1}, y_{m+1}),

$$R_{(m+1)} = R_{(m)} + \bar{x}_{m+1}^{\mathsf{T}} \bar{x}_{m+1}$$

and

$$q_{(m+1)} = q_{(m)} + \bar{x}_{m+1}^{\mathsf{T}} y_{m+1}$$

and the model parameters incorporating information from the new data points satisfy

$$R_{(m+1)}\hat{a}_{(m+1)} = q_{(m+1)} \tag{91}$$

With a known value of a matrix inverse, R^{-1}, the inverse of $R + x^{\mathsf{T}}x$ can be computed via the Sherman-Morrison Update identity

$$(R + x^{\mathsf{T}}x)^{-1} = R^{-1} - R^{-1}x^{\mathsf{T}}\left(1 + xR^{-1}x^{\mathsf{T}}\right)^{-1} xR^{-1} \tag{92}$$

Defining more notation to simplify expressions,

$$K_{(m+1)} = R^{-1}_{(m)} - R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1}(1 + \bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1})^{-1}$$

the rank-1 update of the updated parameter covariance becomes

$$R^{-1}_{(m+1)} = R^{-1}_{(m)} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)} \tag{93}$$

With this we can write $K_{(m+1)}$ as

$$\begin{aligned} K_{(m+1)} &= R^{-1}_{(m+1)}\bar{x}^{\mathsf{T}}_{m+1} \\ &= R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1}\left(1 + \bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1}\right)^{-1} \end{aligned} \tag{94}$$

which can be shown as follows

$$\begin{aligned} K_{(m+1)}\left(1 + \bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1}\right) &= R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} \\ K_{(m+1)} + K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} &= R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} \\ K_{(m+1)} &= R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} \\ &= \left(R^{-1}_{(m)} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\right)\bar{x}^{\mathsf{T}}_{m+1} \\ &= R^{-1}_{(m+1)}\bar{x}^{\mathsf{T}}_{m+1} \end{aligned}$$

Now, introducing a model prediction

$$\hat{y}_{m+1} = \bar{x}_{m+1}\hat{a}_{(m)} = \bar{x}_{m+1}R^{-1}_{(m)}q_{(m)} \tag{95}$$

and combining equations (91) to (95), the update of the model parameter estimates can be written,

$$\hat{a}_{(m+1)} = \hat{a}_{(m)} + K_{(m+1)}\left(y_{m+1} - \hat{y}_{m+1}\right) \tag{96}$$

The role of $K_{(m+1)}$ can be interpreted as an *update gain* which provides the sensitivity of the model parameter update to differences between the measurement y_{m+1} and its predicted value $\hat{y}_{m+1}$. This prediction error is called an *innovation.* The model parameter update identity can be shown as follows:

$$\begin{aligned} \hat{a}_{(m+1)} &= R^{-1}_{(m+1)}q_{(m+1)} \\ &= \left(R^{-1}_{(m)} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\right)\left(q_{(m)} + \bar{x}_{m+1}y_{m+1}\right) \\ &= R^{-1}_{(m)}q_{(m)} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}q_{(m)} + R^{-1}_{(m)}\bar{x}_{m+1}y_{m+1} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\bar{x}_{m+1}y_{m+1} \\ &= \hat{a}_{(m)} - K_{(m+1)}\hat{y}_{m+1} + \left(R^{-1}_{(m)} - K_{(m+1)}\bar{x}_{m+1}R^{-1}_{(m)}\right)\bar{x}_{m+1}y_{m+1} \\ &= \hat{a}_{(m)} - K_{(m+1)}\hat{y}_{m+1} + R^{-1}_{(m+1)}\bar{x}_{m+1}y_{m+1} \\ &= \hat{a}_{(m)} + K_{(m+1)}\left(y_{m+1} - \hat{y}_{m+1}\right) \end{aligned}$$

In many applications of recursive-least squares it is desirable for the most-recent data to have a larger effect upon the parameter estimates. In these situations, the least-squares objective can be exponentially weighted

$$J_{(m)} = \sum_{i=1}^{m} \lambda^{m-i} ||y_i - \hat{y}_i||^2 \qquad 0 \ll \lambda < 1$$

Typical values for the *exponential forgetting factor* λ are close to 1 (e.g., 0.98 to 0.995). Carrying λ through the previous development (91) to (96) we arrive at the recursive least squares procedure

1. initialize variables ... $m = 0$, $R^{-1}_{(m)} = \delta I_n$ where $\delta \geq 100\sigma_x^2$ or $R^{-1}_{(m)} = \left[\sum_{i=-l}^{0} \lambda^{-i} \bar{x}_i^{\mathsf{T}} x_i \right]^{-1}$, and $\hat{a}_{(m)} = 0$ or a knowledgeable guess,
2. collect $\bar{x}_{m+1}$
3. compute the update gain ... $K_{(m+1)} = R^{-1}_{(m)} \bar{x}^{\mathsf{T}}_{m+1} \left(\lambda + \bar{x}_{m+1} R^{-1}_{(m)} \bar{x}^{\mathsf{T}}_{m+1} \right)^{-1}$
4. predict the next measurement ... $\hat{y}_{m+1} = \bar{x}_{m+1} \hat{a}_{(m)}$
5. collect y_{m+1}
6. update the model parameters ... $\hat{a}_{(m+1)} = \hat{a}_{(m)} + K_{(m+1)} \left(y_{m+1} - \hat{y}_{m+1} \right)$
7. update the parameter covariance ... $R^{-1}_{(m+1)} = \left(R^{-1}_{(m)} - K_{(m+1)} \bar{x}_{m+1} R^{-1}_{(m)} \right) / \lambda$
8. increment m ... $m = m + 1$ and go to step 2.

Notes:

- If the update gain is very small, the model parameter update is not sensitive to large prediction errors.
- The update gain decreases monotonically; λ keeps the update gain from becoming too small too fast.
- The update gain increases with larger values of the parameter covariance R^{-1}.
- The update gain can be interpreted as $K \sim V_a/(1 + V_{\hat{y}})$. A large parameter covariance implies large uncertainty in the parameters, and the need for a parameter update that is sensitive to prediction errors (large K). A large model prediction covariance implies noisy data, and the need for a parameter update that is insensitive to prediction errors (small K).
- Likewise, smaller values of λ keep the parameter covariance matrix $R^{-1}_{(m+1)}$ from getting too small too fast.
- With $\lambda = 1$ the RLS estimates $\hat{a}_{(m)}$ equals the OLS estimates $[X^{\mathsf{T}} X]^{-1} X^{\mathsf{T}} y$ obtained from m data points.

13.1 The Sherman-Morrison rank-1 update identity

The Sherman-Morrison Update identity provides the inverse of $(R + x^{\mathsf{T}} x)$ in terms of x and the inverse of R. Defining,

$$R_{(m+1)} = R_{(m)} + x^{\mathsf{T}} x$$

we have

$$\left(R_{(m)} + x^{\mathsf{T}} x\right) R^{-1}_{(m+1)} = I$$

which is equivalent to

$$\left[\begin{array}{cc} R_{(m)} & x^{\mathsf{T}} \\ x & -1 \end{array} \right] \left[\begin{array}{c} R^{-1}_{(m+1)} \\ z \end{array} \right] = \left[\begin{array}{c} I \\ 0 \end{array} \right]$$

To show this, we break the above matrix equation into two separate matrix equations,

$$\begin{aligned} R_{(m)} R^{-1}_{(m+1)} + x^{\mathsf{T}} z &= I \quad (97) \\ x R^{-1}_{(m+1)} - z &= 0 \quad (98) \end{aligned}$$

and substitute the second equation into the first,

$$R_{(m)} R^{-1}_{(m+1)} + x^{\mathsf{T}} x R^{-1}_{(m+1)} = I$$

which shows that

$$\left(R_{(m)} + x^{\mathsf{T}} x\right) R^{-1}_{(m+1)} = I \ .$$

Now re-arraging the first equation,

$$R^{-1}_{(m+1)} = R^{-1}_{(m)} (I - x^{\mathsf{T}} z) \quad (99)$$

and substituting equation (99) into (98) we solve for z,

$$\begin{aligned} z &= x R^{-1}_{(m)} (I - x^{\mathsf{T}} z) \\ &= x R^{-1}_{(m)} - x R^{-1}_{(m)} x^{\mathsf{T}} z \\ \left(1 + x R^{-1}_{(m)} x^{\mathsf{T}}\right) z &= x R^{-1}_{(m)} \\ z &= \left(1 + x R^{-1}_{(m)} x^{\mathsf{T}}\right)^{-1} x R^{-1}_{(m)} \quad (100) \end{aligned}$$

Finally, inserting (100) into (99) we have the Sherman-Morrison Update identity.

$$\begin{aligned} R^{-1}_{(m+1)} &= R^{-1}_{(m)} \left(I - x^{\mathsf{T}} \left(1 + x R^{-1}_{(m)} x^{\mathsf{T}}\right)^{-1} x R^{-1}_{(m)} \right) \\ R^{-1}_{(m+1)} &= R^{-1}_{(m)} - R^{-1}_{(m)} x^{\mathsf{T}} \left(1 + x R^{-1}_{(m)} x^{\mathsf{T}}\right)^{-1} x R^{-1}_{(m)} \end{aligned}$$

13.2 Example of Recursive Least Squares

To illustrate the application of recursive least squares, consider the recursive estimation of parameters a_1 and a_2 in the model

$$\hat{y}(x; a) = a_1 x + a_2 x^2$$

and measurements $y_i = \hat{y}(x_i; a) + v$ with normally-distributed measurement errors, $v \sim \mathcal{N}(0, 0.25)$. The "true" model parameter values are $a_1 = 0.1$ and $a_2 = 0.1$. The sequence of independent variables is $x_m = 0.2m$ so that $\bar{x}_m = [0.2m\ , 0.04m^2]$. The initial parameter covariance is $R_{(0)}^{-1} = I$ and $\lambda = 0.99$ The following figures plot measurements, prediction, and standard errors of the prediction, $\sigma_{\hat{y}} = \left(x_m R_{(m)}^{-1} x_m^{\mathsf{T}} \right)^{1/2}$. These figures show that initially the update gain grows, as the prediction errors covariance is smaller than the parameter covariance. As more data is incorporated into the fit, the parameter covariance and the update gain decrease, as revealed by smaller values of the standard error of the prediction and less fluctuation in the model prediction, despite relatively large prediction errors around $x \approx 8$. At $x \approx 6$ $(m \approx 30)$, the parameters have converted to close to the "true" values.

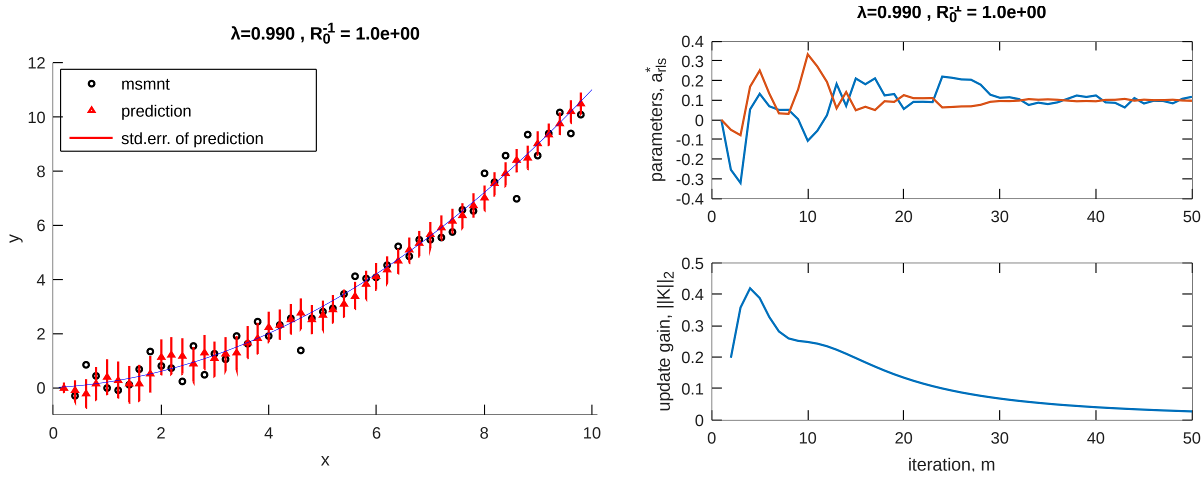

Figure 13. Sequence of predictions and measurements, and sequence of parameter estimates and norms of the update gains.

13.3 Recursive least squares and the Kalman filter

The Kalman filter estimates the states of a noisy dynamical system from a model for the system and noisy output measurements. As is demonstrated below, the Kalman filter can be interpreted as a generalization of recursive least squares for the recursive estimation of the time-varying states of linear dynamical systems. The state vector estimate $\hat{x}(k)$ in the Kalman filter is analogous to the parameter vector estimate $\hat{a}_{(m)}$ in recursive least squares. In recursive least squares, the objective is to recursively converge upon a set of constant model parameters. In Kalman filtering, the objective is to recursively track set of time-varying model states.

A noise-driven discrete-time linear dynamical system with a state vector $x(k) = x(t_k) = x(k(\Delta t))$ and noisy outputs (measurements) $y(k)$ can be described by

$$\begin{array}{rcl} x(k+1) & = & Ax(k) + w(k) \\ y(k) & = & Cx(k) + v(k) \end{array} \tag{101}$$

where $w(k)$ and $v(k)$ are white noise for which the covariance of w is Q and the covariance of v is R. The initial state $x(0)$ is uncertain and assumed to be normally distributed $x(0) \sim \mathcal{N}(\bar{x}(0), P(0))$. The method to sequentially and recursively estimate the state $x(k)$ via the Kalman filter starts by initializing the state vector estimage $\hat{x}(0) = \bar{x}(0)$ and $P(0) = \delta I_n$, and proceeds as follows:

$$K(k+1) = AP(k)C^{\mathsf{T}} \left(CP(k)C^{\mathsf{T}} + R \right)^{-1} \tag{102}$$

$$\hat{y}(k+1) = C\hat{x}(k) \tag{103}$$

$$\hat{x}(k+1) = A\hat{x}(k) + K(k+1)\left(y(k+1) - \hat{y}(k+1)\right) \tag{104}$$

$$P(k+1) = AP(k)A^{\mathsf{T}} - AP(k)C^{\mathsf{T}}(CP(k)C^{\mathsf{T}} + R)^{-1}CP(k)A^{\mathsf{T}} + Q \tag{105}$$

An analogy between the Kalman filter and recursive least squares is tabulated here.

	Kalman filter		recursive least squares
$\hat{x}(k)$	state vector estimate	$\hat{a}_{(m)}$	parameter vector estimate
$\hat{y}(k+1)$	$C\hat{x}(k)$ ... prediction eq'n	$\hat{y}_{m+1}$	$\bar{x}_{m+1}\hat{a}_{(m)}$ ... prediction eq'n
C	output matrix	$\bar{x}_{m+1}$	model basis
$\hat{x}(k+1)$	$A\hat{x}(k) + K(k+1)(y(k+1) - \hat{y}(k+1))$	$\hat{a}_{(m+1)}$	$\hat{a}_{(m)} + K_{(m+1)}(y_{m+1} - \hat{y}_{m+1})$
A	dynamics matrix	I	no dynamics
$K(k+1)$	Kalman gain	$K_{(m+1)}$	update gain
$K(k+1)$	$AP(k)C^{\mathsf{T}} \left(CP(k)C^{\mathsf{T}} + R \right)^{-1}$	$K_{(m+1)}$	$R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} \left(\bar{x}_{m+1}R^{-1}_{(m)}\bar{x}^{\mathsf{T}}_{m+1} + \lambda \right)^{-1}$
$P(m)$	state estimation error covariance	$R^{-1}_{(m)}$	parameter estimation error covariance
R	measurement noise covariance	λ	forgetting factor
Q	additive process noise covariance	$1/\lambda$	multiplicative forgetting factor

In the Kalman filter if $P(0)$ is symmetric then $P(k)$ is symmetric for $k > 0$. Similarly, in recursive least squares, if $R^{-1}_{(0)}$ is symmetric, then $R^{-1}_{(m)}$ remains symmetric for $m > 0$.

References

[1] Forsythe, G.E., "Generation and Use of Orthogonal Polynomials for Data-fitting with a Digital Computer," *J. Soc. Ind. Appl. Math* vol. 5, no 2, 1957.

[2] Hamming, R.W., *Numerical Methods for Scientists and Engineers,* Dover Press, 1986.

[3] Lapin, L.L. *Probability and Statistics for Modern Engineering,* Brooks/Cole, 1983.

[4] Perlis, S., *Theory of Matrices,* Dover Press, 1991.

[5] Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P., *Numerical Recipes,* 2nd ed., Cambridge Univ. Press, 1991.

[6] Tikhonov, A. and Arsin V. *Solutions of Ill Posed Problems,* Wilson and Sons, 1977.

[7] Links related to Inverse Problems (University of Alabama), http://www.me.ua.edu/inverse/