Least Squares Adjustment:
Linear and Nonlinear Weighted Regression Analysis

Allan Aasbjerg Nielsen

Technical University of Denmark
Applied Mathematics and Computer Science/National Space Institute
Building 321, DK-2800 Kgs. Lyngby, Denmark
phone +45 4525 3425, fax +45 4588 1397
http://www.imm.dtu.dk/∼alan
e-mail alan@dtu.dk

19 September 2013

Preface

This note primarily describes the mathematics of least squares regression analysis as it is often used in geodesy including land surveying and satellite based positioning applications. In these fields regression is often termed adjustment[1]. The note also contains a couple of typical land surveying and satellite positioning application examples. In these application areas we are typically interested in the parameters of the model (often 2- or 3-D positions) and their uncertainties and not in predictive modelling which is often the main concern in other regression analysis applications.

Adjustment is often used to obtain estimates of relevant parameters in an over-determined system of equations which may arise from deliberately carrying out more measurements than actually needed to determine the set of desired parameters. An example may be the determination of a geographical position based on information from a number of Global Navigation Satellite System (GNSS) satellites also known as space vehicles (SV). It takes at least four SVs to determine the position (and the clock error) of a GNSS receiver. Often more than four SVs are used and we use adjustment to obtain a better estimate of the geographical position (and the clock error) and to obtain estimates of the uncertainty with which the position is determined.

Regression analysis is used in many other fields of application both in the natural, the technical and the social sciences. Examples may be curve fitting, calibration, establishing relationships between different variables in an experiment or in a survey, etc. Regression analysis is probably one the most used statistical techniques around.

Dr. Anna B. O. Jensen provided insight and data for the Global Positioning System (GPS) example.

Matlab code and sections that are considered as either traditional land surveying material or as advanced material are typeset with smaller fonts.

Comments in general or on for example unavoidable typos, shortcomings and errors are most welcome.

[1] in Danish “udjævning”

Contents

1 Linear Least Squares

Example 1 (from Conradsen, 1984, 1B p. 5.58) Figure 1 shows a plot of clock error as a function of time passed since a calibration of the clock. The relationship between time passed and the clock error seems to be linear (or affine) and it would be interesting to estimate a straight line through the points in the plot, i.e., estimate the slope of the line and the intercept with the axis time = 0. This is a typical regression analysis task (see also Example 2). [end of example]

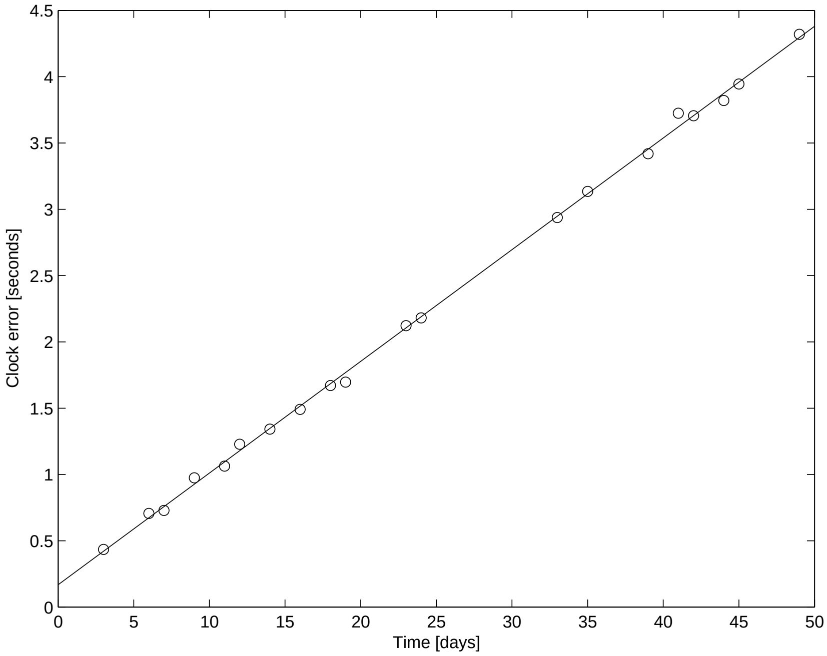

Figure 1: Example with clock error as a function of time.

Let's start by studying a situation where we want to predict one (response) variable y (as clock error in Example 1) as a linear function of one (predictor) variable x (as time in Example 1). When we have one predictor variable only we talk about simple regression. We have n joint observations of x ($x_1, \dots, x_n$) and y ($y_1, \dots, y_n$) and we write the model where the parameter θ_1 is the slope of the line as

$$\begin{aligned} y_1 &= \theta_1 x_1 + e_1 && (1)\\ y_2 &= \theta_1 x_2 + e_2 && (2)\\ &\vdots && (3)\\ y_n &= \theta_1 x_n + e_n. && (4) \end{aligned}$$

The e_is are termed the residuals; they are the differences between the data y_i and the model $\theta_1 x_i$. Rewrite to get

$$\begin{aligned} e_1 &= y_1 - \theta_1 x_1 && (5)\\ e_2 &= y_2 - \theta_1 x_2 && (6)\\ &\vdots && (7)\\ e_n &= y_n - \theta_1 x_n. && (8) \end{aligned}$$

In order to find the best line through (the origo and) the point cloud $\{x_i\ y_i\}_{i=1}^n$ by means of the least squares principle write

$$\epsilon = \frac{1}{2}\sum_{i=1}^{n} e_i^2 = \frac{1}{2}\sum_{i=1}^{n}(y_i - \theta_1 x_i)^2 \tag{9}$$

and find the derivative of ϵ with respect to the slope θ_1

$$\frac{d\epsilon}{d\theta_1} = \sum_{i=1}^{n}(y_i - \theta_1 x_i)(-x_i) = \sum_{i=1}^{n}(\theta_1 x_i^2 - x_i y_i). \tag{10}$$

Setting the derivative equal to zero and denoting the solution $\hat{\theta}_1$ we get

$$\hat{\theta}_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i \tag{11}$$

or (omitting the summation indices for clarity)

$$\hat{\theta}_1 = \frac{\sum x_i y_i}{\sum x_i^2}. \tag{12}$$

Since

$$\frac{d^2\epsilon}{d\theta_1^2} = \sum_{i=1}^{n} x_i^2 > 0 \tag{13}$$

for non-trivial cases $\hat{\theta}_1$ gives a minimum for ϵ. This $\hat{\theta}_1$ gives the best straight line through the origo and the point cloud, “best” in the sense that it minimizes (half) the sum of the squared residuals measured along the y-axis, i.e., perpendicular to the x-axis. In other words: the x_is are considered as uncertainty- or error-free constants, all the uncertainty or error is associated with the y_is.

Let’s look at another situation where we want to predict one (response) variable y as an affine function of one (predictor) variable x. We have n joint observations of x and y and write the model where the parameter θ_0 is the intercept of the line with the y-axis and the parameter θ_1 is the slope of the line as

$$\begin{aligned} y_1 &= \theta_0 + \theta_1 x_1 + e_1 && (14)\\ y_2 &= \theta_0 + \theta_1 x_2 + e_2 && (15)\\ &\vdots && (16)\\ y_n &= \theta_0 + \theta_1 x_n + e_n. && (17) \end{aligned}$$

Rewrite to get

$$\begin{aligned} e_1 &= y_1 - (\theta_0 + \theta_1 x_1) && (18)\\ e_2 &= y_2 - (\theta_0 + \theta_1 x_2) && (19)\\ &\vdots && (20)\\ e_n &= y_n - (\theta_0 + \theta_1 x_n). && (21) \end{aligned}$$

In order to find the best line through the point cloud $\{x_i \ y_i\}_{i=1}^n$ (and this time not necessarily through the origo) by means of the least squares principle write

$$\epsilon = \frac{1}{2}\sum_{i=1}^{n} e_i^2 = \frac{1}{2}\sum_{i=1}^{n}(y_i - (\theta_0 + \theta_1 x_i))^2 \tag{22}$$

and find the partial derivatives of ϵ with respect to the intercept θ_0 and the slope θ_1

$$\frac{\partial\epsilon}{\partial\theta_0} = \sum_{i=1}^{n}(y_i-(\theta_0+\theta_1 x_i))(-1) = -\sum_{i=1}^{n} y_i + n\theta_0 + \theta_1\sum_{i=1}^{n} x_i \tag{23}$$

$$\frac{\partial\epsilon}{\partial\theta_1} = \sum_{i=1}^{n}(y_i-(\theta_0+\theta_1 x_i))(-x_i) = -\sum_{i=1}^{n} x_i y_i + \theta_0\sum_{i=1}^{n} x_i + \theta_1\sum_{i=1}^{n} x_i^2. \tag{24}$$

Setting the partial derivatives equal to zero and denoting the solutions $\hat\theta_0$ and $\hat\theta_1$ we get (omitting the summation indices for clarity)

$$\hat\theta_0 = \frac{\sum x_i^2 \sum y_i - \sum x_i \sum x_i y_i}{n\sum x_i^2 - (\sum x_i)^2} \tag{25}$$

$$\hat\theta_1 = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{n\sum x_i^2 - (\sum x_i)^2}. \tag{26}$$

We see that $\hat\theta_1 \sum x_i + n\hat\theta_0 = \sum y_i$ or $\bar y = \hat\theta_0 + \hat\theta_1 \bar x$ (leading to $\sum \hat e_i = \sum[y_i - (\hat\theta_0 + \hat\theta_1 x_i)] = 0$) where $\bar x = \sum x_i/n$ is the mean value of x and $\bar y = \sum y_i/n$ is the mean value of y. Another way of writing this is

$$\hat\theta_0 = \bar y - \hat\theta_1 \bar x \tag{27}$$

$$\hat\theta_1 = \frac{\sum(x_i-\bar x)(y_i-\bar y)}{\sum(x_i-\bar x)^2} = \frac{\hat\sigma_{xy}}{\hat\sigma_x^2}. \tag{28}$$

where $\hat\sigma_{xy} = \sum(x_i-\bar x)(y_i-\bar y)/(n-1)$ is the covariance between x and y, and $\hat\sigma_x^2 = \sum(x_i-\bar x)^2/(n-1)$ is the variance of x. Also in this case $\hat\theta_0$ and $\hat\theta_1$ give a minimum for ϵ, see page 8.

Example 2 (continuing Example 1) With time points (x_i) [3 6 7 9 11 12 14 16 18 19 23 24 33 35 39 41 42 44 45 49]^T days and clock errors (y_i) [0.435 0.706 0.729 0.975 1.063 1.228 1.342 1.491 1.671 1.696 2.122 2.181 2.938 3.135 3.419 3.724 3.705 3.820 3.945 4.320]^T seconds we get $\hat\theta_0 = 0.1689$ seconds and $\hat\theta_1 = 0.08422$ seconds/day. This line is plotted in Figure 1. Judged visually the line seems to model the data fairly well. [end of example]

More generally let us consider n observations of one dependent (or response) variable y and p' independent (or explanatory or predictor) variables $x_j,\ j = 1,\dots,p'$. The x_js are also called the regressors. When we have more than one regressor we talk about multiple regression analysis. The words "dependent" and "independent" are not used in their probabilistic meaning here but are merely meant to indicate that x_j in principle may vary freely and that y varies depending on x_j. Our task is to 1) estimate the parameters θ_j in the model below, and 2) predict the expectation value of y where we consider y as a function of the θ_js and not of the x_js which are considered as constants. For the ith set of observations we have

$$\begin{aligned} y_i &= y_i(\theta_0,\theta_1,\dots,\theta_{p'};x_1,\dots,x_{p'}) + e_i && (29)\\ &= y_i(\boldsymbol\theta;\boldsymbol x) + e_i && (30)\\ &= y_i(\boldsymbol\theta) + e_i && (31)\\ &= (\theta_0 +)\ \theta_1 x_{i1} + \cdots + \theta_{p'} x_{ip'} + e_i,\ i = 1,\dots,n && (32)\end{aligned}$$

where $\boldsymbol\theta = [\theta_0\ \theta_1\ \dots\ \theta_{p'}]^T$, $\boldsymbol x = [x_1\ \dots\ x_{p'}]^T$, and e_i is the difference between the data and the model for observation i with expectation value $\mathrm{E}\{e_i\} = 0$. e_i is termed the residual or the error. The last equation above is written with the constant or the intercept θ_0 in parenthesis since we may want to include θ_0 in the model or we may not want to, see also Examples 3-5. Write all n equations in matrix form

$$\begin{bmatrix} y_1\\ y_2\\ \vdots\\ y_n\end{bmatrix} = \begin{bmatrix} 1 & x_{11} & \cdots & x_{1p'}\\ 1 & x_{21} & \cdots & x_{2p'}\\ \vdots & \vdots & \ddots & \vdots\\ 1 & x_{n1} & \cdots & x_{np'}\end{bmatrix}\begin{bmatrix}\theta_0\\ \theta_1\\ \vdots\\ \theta_{p'}\end{bmatrix} + \begin{bmatrix} e_1\\ e_2\\ \vdots\\ e_n\end{bmatrix} \tag{33}$$

or

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e} \tag{34}$$

where

- $\boldsymbol{y}$ is $n \times 1$,
- $\boldsymbol{X}$ is $n \times p,\ p = p' + 1$ if an intercept θ_0 is estimated, $p = p'$ if not,
- $\boldsymbol{\theta}$ is $p \times 1$, and
- $\boldsymbol{e}$ is $n \times 1$ with expectation $\mathrm{E}\{\boldsymbol{e}\} = \boldsymbol{0}$.

If we don't want to include θ_0 in the model, θ_0 is omitted from $\boldsymbol{\theta}$ and so is the first column of ones in $\boldsymbol{X}$.

Equations 33 and 34 are termed the observation equations[2]. The columns in $\boldsymbol{X}$ must be linearly independent, i.e., $\boldsymbol{X}$ is full column rank. Here we study the situation where the system of equations is over-determined, i.e., we have more observations than parameters, $n > p$. $f = n - p$ is termed the number of degrees of freedom[3].

The model is linear in the parameters $\boldsymbol{\theta}$ but not necessarily linear in y and x_j (for instance y could be replaced by $\ln y$ or $1/y$, or x_j could be replaced by $\sqrt{x_j}$, extra columns with products $x_k x_l$ called interactions could be added to $\boldsymbol{X}$ or similarly). Transformations of y have implications for the nature of the residual.

Finding an optimal $\boldsymbol{\theta}$ given a set of observed data (the ys and the x_js) and an objective function (or a cost or a merit function, see below) is referred to as regression analysis in statistics. The elements of the vector $\boldsymbol{\theta}$ are also called the regression coefficients. In some application sciences such as geodesy including land surveying regression analysis is termed adjustment[4].

All uncertainty (or error) is associated with y, the x_js are considered as constants which may be reasonable or not depending on (the genesis of) the data to be analyzed.

1.1 Ordinary Least Squares, OLS

In OLS we assume that the variance-covariance matrix also known as the dispersion matrix of $\boldsymbol{y}$ is proportional to the identity matrix, $\mathrm{D}\{\boldsymbol{y}\} = \mathrm{D}\{\boldsymbol{e}\} = \sigma^2\boldsymbol{I}$, i.e., all residuals have the same variance and they are uncorrelated. We minimize the objective function $\epsilon = 1/2\sum_{i=1}^{n} e_i^2 = \boldsymbol{e}^T\boldsymbol{e}/2$ (hence the name least squares: we minimize (half) the sum of squared differences between the data and the model, i.e., (half) the sum of the squared residuals)

$$\begin{aligned}
\epsilon &= 1/2(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^T(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) && (35)\\
&= 1/2(\boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}) && (36)\\
&= 1/2(\boldsymbol{y}^T\boldsymbol{y} - 2\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}). && (37)
\end{aligned}$$

The derivative with respect to $\boldsymbol{\theta}$ is

$$\frac{\partial\epsilon}{\partial\boldsymbol{\theta}} = -\boldsymbol{X}^T\boldsymbol{y} + \boldsymbol{X}^T\boldsymbol{X}\boldsymbol{\theta}. \tag{38}$$

[2] in Danish “observationsligningerne”

[3] in Danish “antal frihedsgrader” or “antal overbestemmelser”

[4] in Danish “udjævning”

When the columns of $\boldsymbol{X}$ are linearly independent the second order derivative $\partial^2\epsilon/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T = \boldsymbol{X}^T\boldsymbol{X}$ is positive definite. Therefore we have a minimum for ϵ. Note that the $p \times p$ $\boldsymbol{X}^T\boldsymbol{X}$ is symmetric, $(\boldsymbol{X}^T\boldsymbol{X})^T = \boldsymbol{X}^T\boldsymbol{X}$.

We find the OLS estimate for $\boldsymbol{\theta}$ termed $\hat{\boldsymbol{\theta}}_{OLS}$ (pronounced theta-hat) by setting $\partial\epsilon/\partial\boldsymbol{\theta} = \boldsymbol{0}$ to obtain the normal equations[5]

$$\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{X}^T\boldsymbol{y}. \tag{39}$$

1.1.1 Linear Constraints

Linear constraints can be build into the normal equations by defining

$$\boldsymbol{K}^T\boldsymbol{\theta} \;=\; \boldsymbol{c} \tag{40}$$

where the vector $\boldsymbol{c}$ and the columns of matrix $\boldsymbol{K}$ define the constraints, one constraint per column of $\boldsymbol{K}$ and per element of $\boldsymbol{c}$. If for example $\boldsymbol{\theta} = [\theta_1\ \theta_2\ \theta_3\ \theta_4\ \theta_5]^T$ and θ_2, θ_3 and θ_5 are the three angles in a triangle which must sum to π—also known as 200 gon in land surveying—(with no constraints on θ_1 and θ_4), use $\boldsymbol{K}^T = [0\ 1\ 1\ 0\ 1]$ and $\boldsymbol{c} = 200$ gon.

Also, we must add a term to the expression for ϵ in Equation 35 above setting the constraints to zero

$$L \;=\; \epsilon + \boldsymbol{\lambda}^T(\boldsymbol{K}^T\boldsymbol{\theta} - \boldsymbol{c}) \tag{41}$$

where $\boldsymbol{\lambda}$ is a vector of so-called Lagrangian multipliers.

Setting the partial derivatives of Equations 41 and 40 to zero leads to

$$\begin{bmatrix} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{K} \\ \boldsymbol{K}^T & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\theta}}_{OLS} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}^T\boldsymbol{y} \\ \boldsymbol{c} \end{bmatrix}. \tag{42}$$

1.1.2 Parameter Estimates

If the symmetric matrix $\boldsymbol{X}^T\boldsymbol{X}$ is "well behaved", i.e., it is full rank (equal to p) corresponding to linearly independent columns in $\boldsymbol{X}$ a formal solution is

$$\hat{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\,\boldsymbol{y}. \tag{43}$$

For reasons of numerical stability especially in situations with nearly linear dependencies between the columns of $\boldsymbol{X}$ (causing slight alterations to the observed values in $\boldsymbol{X}$ to lead to substantial changes in the estimated $\hat{\boldsymbol{\theta}}$; this problem is known as multicollinearity) the system of normal equations should not be solved by inverting $\boldsymbol{X}^T\boldsymbol{X}$ but rather by means of SVD, QR or Cholesky decomposition, see Sections 1.1.6, 1.1.7 and 1.1.8.

If we apply Equation 43 to the simple regression problem in Equations 14-17 of course we get the same solution as in Equations 25 and 26 (as an exercise you may want to check this).

When we apply regression analysis in other application areas we are often interested in predicting the response variable based on new data not used in the estimation of the parameters or the regression coefficients $\hat{\boldsymbol{\theta}}$. In land surveying and GNSS applications we are typically interested in $\hat{\boldsymbol{\theta}}$ and not on this predictive modelling.

(In the linear case $\hat{\boldsymbol{\theta}}_{OLS}$ can be found in one go because $\boldsymbol{e}^T\boldsymbol{e}$ is quadratic in $\boldsymbol{\theta}$; unlike in the nonlinear case dealt with in Section 2 we don't need an initial value for $\boldsymbol{\theta}$ and an iterative procedure.)

[5] in Danish "normalligningerne"

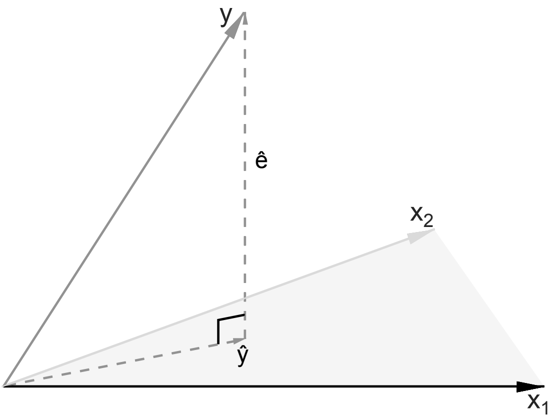

Figure 2: $\hat{\boldsymbol{y}}$ is the projection of $\boldsymbol{y}$ onto the hyperplane spanned by the vectors $\boldsymbol{x}_i$ in the columns of matrix $\boldsymbol{X}$ (modified from Hastie, Tibshirani and Friedman (2009) by Jacob S. Vestergaard).

The estimate for $\boldsymbol{y}$ termed $\hat{\boldsymbol{y}}$ (pronounced y-hat) is

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} = \boldsymbol{H}\boldsymbol{y} \tag{44}$$

where $\boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$ is the so-called hat matrix since it transforms or projects $\boldsymbol{y}$ into $\hat{\boldsymbol{y}}$ ($\boldsymbol{H}$ "puts the hat on $\boldsymbol{y}$"). In geodesy (and land surveying) these equations are termed the fundamental equations[6]. $\boldsymbol{H}$ is a projection matrix: it is symmetric, $\boldsymbol{H} = \boldsymbol{H}^T$, and idempotent, $\boldsymbol{H}\boldsymbol{H} = \boldsymbol{H}$. We also have $\boldsymbol{H}\boldsymbol{X} = \boldsymbol{X}$ and that the trace of $\boldsymbol{H}$, tr$\boldsymbol{H}$ = tr$(\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T)$ = tr$(\boldsymbol{X}^T\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1})$ = tr$\boldsymbol{I}_p = p$.

The estimate of the error term $\boldsymbol{e}$ (also known as the residual) termed $\hat{\boldsymbol{e}}$ (pronounced e-hat) is

$$\hat{\boldsymbol{e}} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{H}\boldsymbol{y} = (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y}. \tag{45}$$

Also $\boldsymbol{I} - \boldsymbol{H}$ is symmetric, $\boldsymbol{I} - \boldsymbol{H} = (\boldsymbol{I} - \boldsymbol{H})^T$, and idempotent, $(\boldsymbol{I} - \boldsymbol{H})(\boldsymbol{I} - \boldsymbol{H}) = \boldsymbol{I} - \boldsymbol{H}$. We also have $(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{X} = \boldsymbol{0}$ and tr$(\boldsymbol{I} - \boldsymbol{H}) = n - p$.

$\boldsymbol{X}$ and $\hat{\boldsymbol{e}}$, and $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}$ are orthogonal: $\boldsymbol{X}^T\hat{\boldsymbol{e}} = \boldsymbol{0}$ and $\hat{\boldsymbol{y}}^T\hat{\boldsymbol{e}} = 0$. Geometrically this means that our analysis finds the orthogonal projection $\hat{\boldsymbol{y}}$ of $\boldsymbol{y}$ onto the hyperplane spanned by the linearly independent columns of $\boldsymbol{X}$. this gives the shortest distance between $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$, see Figure 2.

Since the expectation of $\hat{\boldsymbol{\theta}}_{OLS}$

$$\begin{aligned}
\mathrm{E}\{\hat{\boldsymbol{\theta}}_{OLS}\} &= \mathrm{E}\{(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}\} \qquad (46)\\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathrm{E}\{\boldsymbol{y}\} \qquad (47)\\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathrm{E}\{\boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e}\} \qquad (48)\\
&= \boldsymbol{\theta}, \qquad (49)
\end{aligned}$$

$\hat{\boldsymbol{\theta}}_{OLS}$ is unbiased or a central estimator.

1.1.3 Regularization

If $\boldsymbol{X}^T\boldsymbol{X}$ is near singular (also known as ill-conditioned) we may use so-called regularization. In the regularized case we penalize some characteristic of $\boldsymbol{\theta}$, for example size, by introducing an extra term into Equation 35 (typically with $\boldsymbol{X}^T\boldsymbol{X}$ normalized to correlation form), namely $\lambda\boldsymbol{\theta}^T\boldsymbol{\Omega}\boldsymbol{\theta}$ where $\boldsymbol{\Omega}$ describes some characteristic of $\boldsymbol{\theta}$ and the small positive scalar λ determines the

[6] in Danish "fundamentalligningerne"

amount of regularization. If we wish to penalize large θ_i, i.e., we wish to penalize size, $\boldsymbol{\Omega}$ is the unit matrix. In this case we use the term ridge regression. In the regularized case the normal equations become

$$(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{\Omega})\tilde{\boldsymbol{\theta}}_{OLS} = \boldsymbol{X}^T\boldsymbol{y}, \tag{50}$$

with formal solution

$$\tilde{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{\Omega})^{-1}\boldsymbol{X}^T\boldsymbol{y}. \tag{51}$$

For ridge regression this becomes

$$\tilde{\boldsymbol{\theta}}_{OLS} = (\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T\boldsymbol{y} = (\boldsymbol{I} + \lambda(\boldsymbol{X}^T\boldsymbol{X})^{-1})^{-1}\hat{\boldsymbol{\theta}}_{OLS}. \tag{52}$$

Example 3 (from Strang and Borre, 1997, p. 306) Between four points A, B, C and D situated on a straight line we have measured all pairwise distances AB, BC, CD, AC, AD and BD. The six measurements are $\boldsymbol{y} = [3.17\ 1.12\ 2.25\ 4.31\ 6.51\ 3.36]^T$ m. We wish to determine the distances $\theta_1 = AB$, $\theta_2 = BC$ and $\theta_3 = CD$ by means of linear least squares adjustment. We have $n = 6$, $p = 3$ and $f = 3$. The six observation equations are

$$\begin{aligned} y_1 &= \theta_1 + e_1 & (53)\\ y_2 &= \theta_2 + e_2 & (54)\\ y_3 &= \theta_3 + e_3 & (55)\\ y_4 &= \theta_1 + \theta_2 + e_4 & (56)\\ y_5 &= \theta_1 + \theta_2 + \theta_3 + e_5 & (57)\\ y_6 &= \theta_2 + \theta_3 + e_6. & (58) \end{aligned}$$

In matrix form we get (this is $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e}$; units are m)

$$\begin{bmatrix} 3.17 \\ 1.12 \\ 2.25 \\ 4.31 \\ 6.51 \\ 3.36 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix}. \tag{59}$$

The normal equations are (this is $\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}} = \boldsymbol{X}^T\boldsymbol{y}$; units are m)

$$\begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 13.99 \\ 15.30 \\ 12.12 \end{bmatrix}. \tag{60}$$

The hat matrix is

$$\boldsymbol{H} = \begin{bmatrix} 1/2 & -1/4 & 0 & 1/4 & 1/4 & -1/4 \\ -1/4 & 1/2 & -1/4 & 1/4 & 0 & 1/4 \\ 0 & -1/4 & 1/2 & -1/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & -1/4 & 1/2 & 1/4 & 0 \\ 1/4 & 0 & 1/4 & 1/4 & 1/2 & 1/4 \\ -1/4 & 1/4 & 1/4 & 0 & 1/4 & 1/2 \end{bmatrix}. \tag{61}$$

The solution is $\hat{\boldsymbol{\theta}} = [3.1700\ 1.1225\ 2.2350]^T$ m, see Matlab code in page 13.

Now, let us estimate an intercept θ_0 also corresponding to an imprecise zero mark of the distance measuring device used. In this case we have $n = 6$, $p = 4$ and $f = 2$ and we get (in m)

$$\begin{bmatrix} 3.17 \\ 1.12 \\ 2.25 \\ 4.31 \\ 6.51 \\ 3.36 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix}. \tag{62}$$

The normal equations in this case are (in m)

$$\begin{bmatrix} 6 & 3 & 4 & 3 \\ 3 & 3 & 2 & 1 \\ 4 & 2 & 4 & 2 \\ 3 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 20.72 \\ 13.99 \\ 15.30 \\ 12.12 \end{bmatrix}. \tag{63}$$

The hat matrix is

$$\boldsymbol{H} = \begin{bmatrix} 3/4 & 0 & 1/4 & 1/4 & 0 & -1/4 \\ 0 & 3/4 & 0 & 1/4 & -1/4 & 1/4 \\ 1/4 & 0 & 3/4 & -1/4 & 0 & 1/4 \\ 1/4 & 1/4 & -1/4 & 1/2 & 1/4 & 0 \\ 0 & -1/4 & 0 & 1/4 & 3/4 & 1/4 \\ -1/4 & 1/4 & 1/4 & 0 & 1/4 & 1/2 \end{bmatrix}. \tag{64}$$

The solution is $\hat{\boldsymbol{\theta}} = [0.0150\ 3.1625\ 1.1150\ 2.2275]^T$ m, see Matlab code in page 13. [end of example]

1.1.4 Dispersion and Significance of Estimates

Dispersion or variance-covariance matrices for $\boldsymbol{y}$, $\hat{\boldsymbol{\theta}}_{OLS}$, $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}$ are

$$\begin{aligned}
\mathrm{D}\{\boldsymbol{y}\} &= \sigma^2 \boldsymbol{I} && (65)\\
\mathrm{D}\{\hat{\boldsymbol{\theta}}_{OLS}\} &= \mathrm{D}\{(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}\} && (66)\\
&= (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\mathrm{D}\{\boldsymbol{y}\}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1} && (67)\\
&= \sigma^2(\boldsymbol{X}^T\boldsymbol{X})^{-1} && (68)\\
\mathrm{D}\{\hat{\boldsymbol{y}}\} &= \mathrm{D}\{\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS}\} && (69)\\
&= \boldsymbol{X}\mathrm{D}\{\hat{\boldsymbol{\theta}}_{OLS}\}\boldsymbol{X}^T && (70)\\
&= \sigma^2\boldsymbol{H},\ \mathrm{V}\{\hat{y}_i\} = \sigma^2 H_{ii} && (71)\\
\mathrm{D}\{\hat{\boldsymbol{e}}\} &= \mathrm{D}\{(\boldsymbol{I}-\boldsymbol{H})\boldsymbol{y}\} && (72)\\
&= (\boldsymbol{I}-\boldsymbol{H})\mathrm{D}\{\boldsymbol{y}\}(\boldsymbol{I}-\boldsymbol{H})^T && (73)\\
&= \sigma^2(\boldsymbol{I}-\boldsymbol{H}) = \mathrm{D}\{\boldsymbol{y}\} - \mathrm{D}\{\hat{\boldsymbol{y}}\},\ \mathrm{V}\{\hat{e}_i\} = \sigma^2(1-H_{ii}). && (74)
\end{aligned}$$

The ith diagonal element of $\boldsymbol{H}$, H_{ii}, is called the leverage[7] for observation i. We see that a high leverage gives a high variance for $\hat{y}_i$ indicating that observation i is poorly predicted by the regression model. This again indicates that observation i may be an outlier, see also Section 1.1.5 on residual and influence analysis.

[7] in Danish "potentialet"

For the sum of squared errors (SSE, also called RSS for the residual sum of squares) we get

$$\hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}} = \boldsymbol{y}^T(\boldsymbol{I}-\boldsymbol{H})\boldsymbol{y} \tag{75}$$

with expectation $\mathrm{E}\{\hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}\} = \sigma^2(n-p)$. The mean squared error MSE is

$$\hat{\sigma}^2 = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}/(n-p) \tag{76}$$

and the root mean squared error RMSE is $\hat{\sigma}$ also known as s. $\hat{\sigma} = s$ has the same unit as e_i and y_i.

The square roots of the diagonal elements of the dispersion matrices in Equations 65, 68, 71 and 74 are the standard errors of the quantities in question. For example, the standard error of $\hat{\theta}_i$ denoted $\hat{\sigma}_{\theta_i}$ is the square root of the ith diagonal element of $\sigma^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}$.

Example 4 (continuing Example 3) The estimated residuals in the case with no intercept are $\hat{\boldsymbol{e}} = [0.0000\ {-0.0025}\ 0.0150\ 0.0175\ {-0.0175}\ 0.0025]^T$ m. Therefore the RMSE or $\hat{\sigma} = s = \sqrt{\hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}/3}$ m $= 0.0168$ m. The inverse of $\boldsymbol{X}^T\boldsymbol{X}$ is

$$\begin{bmatrix} 3 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}^{-1} = \begin{bmatrix} 1/2 & -1/4 & 0 \\ -1/4 & 1/2 & -1/4 \\ 0 & -1/4 & 1/2 \end{bmatrix}. \tag{77}$$

This gives standard deviations for $\boldsymbol{\theta}$, $\hat{\sigma}_\theta = [0.0119\ 0.0119\ 0.0119]^T$ m. The case with an intercept gives $\hat{\sigma} = s = 0.0177$ m and standard deviations for $\boldsymbol{\theta}$, $\hat{\sigma}_\theta = [0.0177\ 0.0153\ 0.0153\ 0.0153]^T$ m. [end of example]

So far we have assumed only that $\mathrm{E}\{\boldsymbol{e}\} = \boldsymbol{0}$ and that $\mathrm{D}\{\boldsymbol{e}\} = \sigma^2\boldsymbol{I}$, i.e., we have made no assumptions about the distribution of $\boldsymbol{e}$. Let us further assume that the e_is are independent and identically distributed (written as iid) following a normal distribution. Then $\hat{\boldsymbol{\theta}}_{OLS}$ (which in this case corresponds to a maximum likelihood estimate) follows a multivariate normal distribution with mean $\boldsymbol{\theta}$ and dispersion $\sigma^2(\boldsymbol{X}^T\boldsymbol{X})^{-1}$. Assuming that $\hat{\theta}_i = c_i$ where c_i is a constant it can be shown that the ratio

$$z_i = \frac{\hat{\theta}_i - c_i}{\hat{\sigma}_{\theta_i}} \tag{78}$$

follows a t distribution with $n-p$ degrees of freedom. This can be used to test whether $\hat{\theta}_i - c_i$ is significantly different from 0. If for example z_i with $c_i = 0$ has a small absolute value then $\hat{\theta}_i$ is not significantly different from 0 and x_i should be removed from the model.

Example 5 (continuing Example 4) The t-test statistics z_i with $c_i = 0$ in the case with no intercept are $[266.3\ 94.31\ 187.8]^T$ which are all very large compared to 95% or 99% percentiles in a two-sided t-test with three degrees of freedom, 3.182 and 5.841 respectively. The probabilities of finding larger values of $|z_i|$ are $[0.0000\ 0.0000\ 0.0000]^T$. Hence all parameter estimates are significantly different from zero. The t-test statistics z_i with $c_i = 0$ in the case with an intercept are $[0.8485\ 206.6\ 72.83\ 145.5]^T$; all but the first value are very large compared to 95% and 99% percentiles in a two-sided t-test with two degrees of freedom, 4.303 and 9.925 respectively. The probabilities of finding larger values of $|z_i|$ are $[0.4855\ 0.0000\ 0.0002\ 0.0000]^T$. Therefore the estimate of θ_0 is insignificant (i.e., it is not significantly different from zero) and the intercept corresponding to an imprecise zero mark of the distance measuring device used should not be included in the model. [end of example]

Often a measure of variance reduction termed the coefficient of determination denoted R^2 and a version that adjusts for the number of parameters denoted R^2_{adj} are defined in the statistical literature:

$\mathrm{SST}_0 = \boldsymbol{y}^T\boldsymbol{y}$ (if no intercept θ_0 is estimated)
$\mathrm{SST}_1 = (\boldsymbol{y}-\bar{\boldsymbol{y}})^T(\boldsymbol{y}-\bar{\boldsymbol{y}})$ (if an intercept θ_0 is estimated)
$\mathrm{SSE} = \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}$
$R^2 = 1 - \mathrm{SSE}/\mathrm{SST}_i$
$R^2_{adj} = 1 - (1-R^2)(n-i)/(n-p)$ where i is 0 or 1 as indicated by SST_i.

Both R^2 and R^2_{adj} lie in the interval [0,1]. For a good model with a good fit to the data both R^2 and R^2_{adj} should be close to 1.

Matlab code for Examples 3 to 5

```
% (C) Copyright 2003
% Allan Aasbjerg Nielsen
% aa@imm.dtu.dk, www.imm.dtu.dk/˜aa

% model without intercept

y = [3.17 1.12 2.25 4.31 6.51 3.36]';
X = [1 0 0; 0 1 0; 0 0 1; 1 1 0; 1 1 1; 0 1 1];
[n,p] = size(X);
f = n-p;

thetah = X\y;
yh = X*thetah;
eh = y-yh;
s2 = eh'*eh/f;
s = sqrt(s2);
iXX = inv(X'*X);
Dthetah = s2.*iXX;
stdthetah = sqrt(diag(Dthetah));
t = thetah./stdthetah;
pt = betainc(f./(f+t.^2),0.5*f,0.5);

H = X*iXX*X';
Hii = diag(H);

% model with intercept

X = [ones(n,1) X];
[n,p] = size(X);
f = n-p;

thetah = X\y;
yh = X*thetah;
eh = y-yh;
s2 = eh'*eh/f;
s = sqrt(s2);
iXX = inv(X'*X);
Dthetah = s2.*iXX;
stdthetah = sqrt(diag(Dthetah));
t = thetah./stdthetah;
pt = betainc(f./(f+t.^2),0.5*f,0.5);

H = X*iXX*X';
Hii = diag(H);
```

The Matlab backslash operator "\" or `mldivide`, "left matrix divide", in this case with $\boldsymbol{X}$ non-square computes the QR factorization (see Section 1.1.7) of $\boldsymbol{X}$ and finds the least squares solution by back-substitution.

Probabilities in the t distribution are calculated by means of the incomplete beta function evaluated in Matlab by the `betainc` function.

1.1.5 Residual and Influence Analysis

Residual analysis is performed to check the model and to find possible outliers or gross errors in the data. Often inspection of listings or plots of $\hat{\boldsymbol{e}}$ against $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}$ against the columns in $\boldsymbol{X}$ (the explanatory variables or the regressors) are useful. No systematic tendencies should be observable in these listings or plots.

Standardized residuals

$$e_i' = \frac{\hat{e}_i}{\hat{\sigma}\sqrt{1 - H_{ii}}} \tag{79}$$

which have unit variance (see Equation 74) are often used.

Studentized or jackknifed residuals (regression omitting observation i to obtain a prediction for the omitted observation $\hat{y}_{(i)}$ and an estimate of the corresponding error variance $\hat{\sigma}^2_{(i)}$)

$$e_i^* = \frac{y_i - \hat{y}_{(i)}}{\sqrt{\mathrm{V}\{y_i - \hat{y}_{(i)}\}}} \tag{80}$$

are also often used. We don't have to redo the adjustment each time an observation is left out since it can be shown that

$$e_i^* = e_i' \bigg/ \sqrt{\frac{n-p-e_i'^2}{n-p-1}}\,. \tag{81}$$

For the sum of the diagonal elements H_{ii} of the hat matrix we have tr$\boldsymbol{H} = \sum_{i=1}^n H_{ii} = p$ which means that the average value $\bar{H}_{ii} = p/n$. Therefore an alarm for very influential observations which may be outliers could be set if $H_{ii} > 2p/n$ (or maybe if $H_{ii} > 3p/n$). As mentioned above H_{ii} is termed the leverage for observation i. None of the observations in Example 3 have high leverages.

Another often used measure of influence of the individual observations is called Cook's distance also known as Cook's D. Cook's D for observation i measures the distance between the vector of estimated parameters with and without observation i (often skipping the intercept $\hat{\theta}_0$ if estimated). Other influence statistics exist.

Example 6 In this example two data sets are simulated. The first data set contains 100 observations with one outlier. This outlier is detected by means of its residual, the leverage of the outlier is low since the observation does not influence the regression line, see Figure 3. In the top-left panel the dashed line is from a regression with an insignificant intercept and the solid line is from a regression without the intercept. The outlier has a huge residual, see the bottom-left panel. The mean leverage is $p/n = 0.01$. Only a few leverages are greater then 0.02, see the top-right panel. No leverages are greater then 0.03.

The second data set contains four observations with one outlier, see Figure 3 bottom-right panel. This outlier (observation 4 with coordinates (100,10)) is detected by means of its leverage, the residual of the outlier is low, see Table 1. The mean leverage is $p/n = 0.5$. The leverage of the outlier is by far the greatest, $H_{44} \simeq 2p/n$. [end of example]

1.1.6 Singular Value Decomposition, SVD

In general the data matrix $\boldsymbol{X}$ can be factorized as

$$\boldsymbol{X} = \boldsymbol{V}\boldsymbol{\Gamma}\boldsymbol{U}^T, \tag{82}$$

Table 1: Residuals and leverages for simulated example with one outlier (observation 4) detected by the leverage.

Obs	x	y	Residual	Leverage
1	1	1	–0.9119	0.3402
2	2	2	0.0062	0.3333
3	3	3	0.9244	0.3266
4	100	10	–0.0187	0.9998

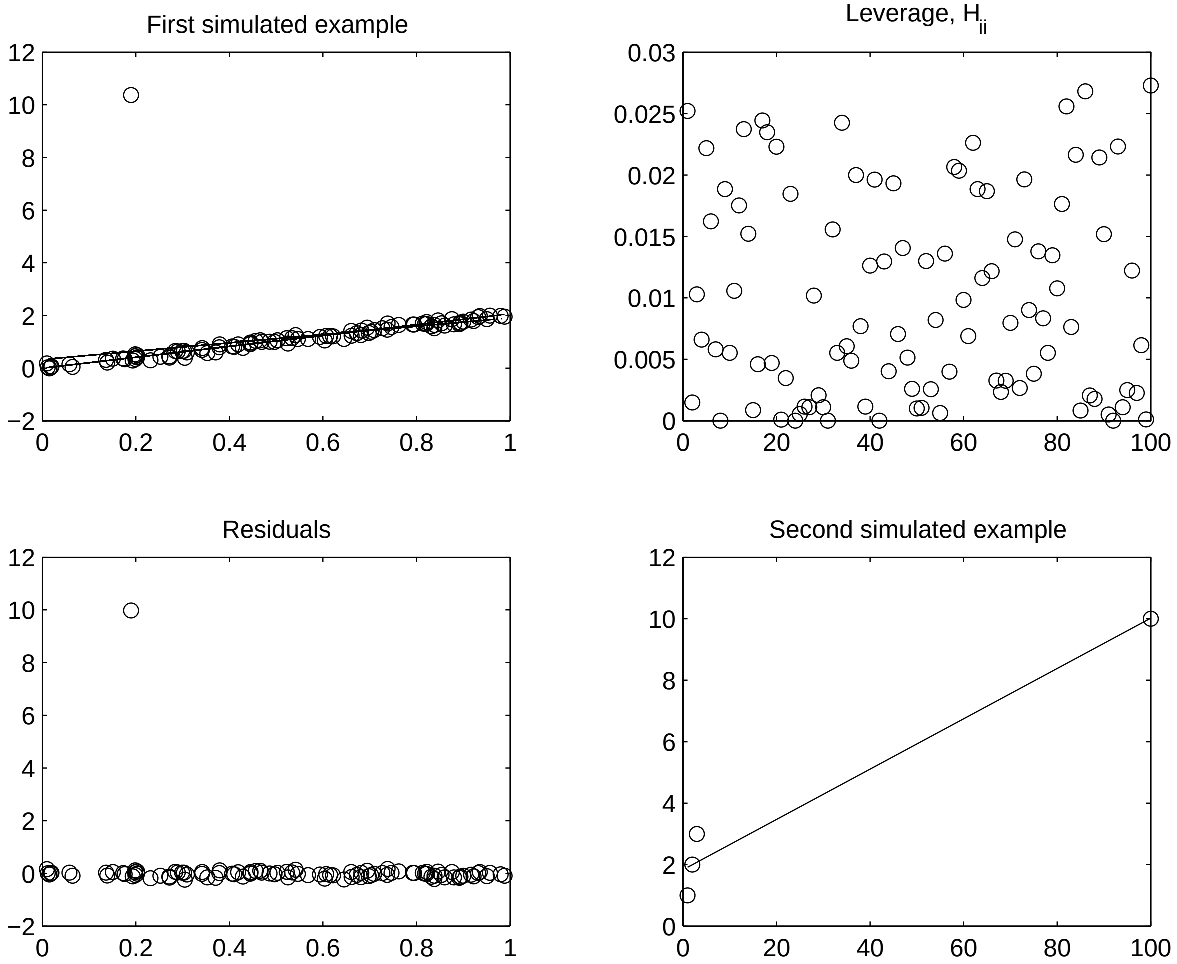

Figure 3: Simulated examples with 1) one outlier detected by the residual (top-left and bottom-left) and 2) one outlier (observation 4) detected by the leverage (bottom-right).

where $\boldsymbol{V}$ is $n \times p$, $\boldsymbol{\Gamma}$ is $p \times p$ diagonal with the singular values of $\boldsymbol{X}$ on the diagonal, and $\boldsymbol{U}$ is $p \times p$ with $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}_p$. This leads to the following solution to the normal equations

$$\begin{aligned} \boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{X}^T\boldsymbol{y} && (83)\\ (\boldsymbol{V}\boldsymbol{\Gamma}\boldsymbol{U}^T)^T(\boldsymbol{V}\boldsymbol{\Gamma}\boldsymbol{U}^T)\hat{\boldsymbol{\theta}}_{OLS} &= (\boldsymbol{V}\boldsymbol{\Gamma}\boldsymbol{U}^T)^T\boldsymbol{y} && (84)\\ \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{\Gamma}\boldsymbol{U}^T\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{V}^T\boldsymbol{y} && (85)\\ \boldsymbol{U}\boldsymbol{\Gamma}^2\boldsymbol{U}^T\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{U}\boldsymbol{\Gamma}\boldsymbol{V}^T\boldsymbol{y} && (86)\\ \boldsymbol{\Gamma}\boldsymbol{U}^T\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{V}^T\boldsymbol{y} && (87) \end{aligned}$$

and therefore

$$\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{U}\boldsymbol{\Gamma}^{-1}\boldsymbol{V}^T\boldsymbol{y}. \qquad (88)$$

1.1.7 QR Decomposition

An alternative factorization of $\boldsymbol{X}$ is

$$\boldsymbol{X} = \boldsymbol{Q}\boldsymbol{R}, \qquad (89)$$

where $\boldsymbol{Q}$ is $n \times p$ with $\boldsymbol{Q}^T\boldsymbol{Q} = \boldsymbol{I}_p$ and $\boldsymbol{R}$ is $p \times p$ upper triangular. This leads to

$$\boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{X}^T\boldsymbol{y} \qquad (90)$$

$$\begin{aligned} (\boldsymbol{QR})^T\boldsymbol{QR}\hat{\boldsymbol{\theta}}_{OLS} &= (\boldsymbol{QR})^T\boldsymbol{y} && (91)\\ \boldsymbol{R}^T\boldsymbol{Q}^T\boldsymbol{QR}\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{R}^T\boldsymbol{Q}^T\boldsymbol{y} && (92)\\ \boldsymbol{R}\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{Q}^T\boldsymbol{y}. && (93)\end{aligned}$$

This system of equations can be solved by back-substitution.

1.1.8 Cholesky Decomposition

Both the SVD and the QR factorizations work on $\boldsymbol{X}$. Here we factorize $\boldsymbol{X}^T\boldsymbol{X}$

$$\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{C}\boldsymbol{C}^T, \tag{94}$$

where $\boldsymbol{C}$ is $p \times p$ lower triangular. This leads to

$$\begin{aligned} \boldsymbol{X}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{X}^T\boldsymbol{y} && (95)\\ \boldsymbol{C}\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{X}^T\boldsymbol{y}. && (96)\end{aligned}$$

This system of equations can be solved by two times back-substitution.

A Trick to Obtain $\sqrt{\hat{e}^T\hat{e}}$ with the Cholesky Decomposition $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{C}\boldsymbol{C}^T$, $\boldsymbol{C}$ is $p \times p$ lower triangular

$$\begin{aligned} \boldsymbol{C}\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS} &= \boldsymbol{X}^T\boldsymbol{y} && (97)\\ \boldsymbol{C}(\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS}) &= \boldsymbol{X}^T\boldsymbol{y} && (98)\end{aligned}$$

so $\boldsymbol{C}\boldsymbol{z} = \boldsymbol{X}^T\boldsymbol{y}$ with $\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS} = \boldsymbol{z}$. Expand $p \times p$ $\boldsymbol{X}^T\boldsymbol{X}$ with one more row and column to $(p+1) \times (p+1)$

$$\tilde{\boldsymbol{C}}\tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{X}^T\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{y} \\ (\boldsymbol{X}^T\boldsymbol{y})^T & \boldsymbol{y}^T\boldsymbol{y} \end{bmatrix}. \tag{99}$$

With

$$\tilde{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \\ \boldsymbol{z}^T & s \end{bmatrix} \text{ and } \tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{C}^T & \boldsymbol{z} \\ \boldsymbol{0}^T & s \end{bmatrix} \tag{100}$$

we get

$$\tilde{\boldsymbol{C}}\tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{C}\boldsymbol{C}^T & \boldsymbol{C}\boldsymbol{z} \\ \boldsymbol{z}^T\boldsymbol{C}^T & \boldsymbol{z}^T\boldsymbol{z}+s^2 \end{bmatrix}. \tag{101}$$

We see that

$$\begin{aligned} s^2 &= \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{z}^T\boldsymbol{z} && (102)\\ &= \boldsymbol{y}^T\boldsymbol{y} - \hat{\boldsymbol{\theta}}^T_{OLS}\boldsymbol{C}\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS} && (103)\\ &= \boldsymbol{y}^T\boldsymbol{y} - \hat{\boldsymbol{\theta}}^T_{OLS}\boldsymbol{X}^T\boldsymbol{y} && (104)\\ &= \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{X}\hat{\boldsymbol{\theta}}_{OLS} && (105)\\ &= \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y} && (106)\\ &= \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{H}\boldsymbol{y} && (107)\\ &= \boldsymbol{y}^T(\boldsymbol{I}-\boldsymbol{H})\boldsymbol{y} && (108)\\ &= \hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}. && (109)\end{aligned}$$

Hence, after Cholesky decomposition of the expanded matrix, the lower right element of $\tilde{\boldsymbol{C}}$ is $\sqrt{\hat{\boldsymbol{e}}^T\hat{\boldsymbol{e}}}$. The last column in $\tilde{\boldsymbol{C}}^T$ (skipping s in the last row) is $\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{OLS}$, hence $\hat{\boldsymbol{\theta}}_{OLS}$ can be found by back-substitution.

1.2 Weighted Least Squares, WLS

In WLS we allow the uncorrelated residuals to have different variances and assume that $\text{D}\{\boldsymbol{y}\} = \text{D}\{\boldsymbol{e}\} = \text{diag}[\sigma_1^2, \ldots, \sigma_n^2]$. We assign a weight p_i (p for pondus which is Latin for weight) to each observation so that $p_1\sigma_1^2 = \cdots = p_i\sigma_i^2 = \cdots = p_n\sigma_n^2 = 1 \cdot \sigma_0^2$ or $\sigma_i^2 = \sigma_0^2/p_i$ with $p_i > 0$. σ_0 is termed the standard deviation of unit weight[8]. Therefore $\text{D}\{\boldsymbol{y}\} = \text{D}\{\boldsymbol{e}\} = \sigma_0^2\, \text{diag}[1/p_1, \ldots, 1/p_n] = \sigma_0^2\, \boldsymbol{P}^{-1}$ and we minimize the objective function $\epsilon = 1/2 \sum_{i=1}^n p_i e_i^2 = \boldsymbol{e}^T\boldsymbol{P}\boldsymbol{e}/2$ where

$$\boldsymbol{P} = \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ 0 & p_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p_n \end{bmatrix}. \tag{110}$$

We get

$$\begin{aligned} \epsilon &= 1/2(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^T\boldsymbol{P}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) & (111)\\ &= 1/2(\boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\boldsymbol{\theta}) & (112)\\ &= 1/2(\boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - 2\boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\boldsymbol{\theta}). & (113) \end{aligned}$$

The derivative with respect to $\boldsymbol{\theta}$ is

$$\frac{\partial\epsilon}{\partial\boldsymbol{\theta}} = -\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} + \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\boldsymbol{\theta}. \tag{114}$$

When the columns of $\boldsymbol{X}$ are linearly independent the second order derivative $\partial^2\epsilon/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ is positive definite. Therefore we have a minimum for ϵ. Note that $\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ is symmetric, $(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^T = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$.

We find the WLS estimate for $\boldsymbol{\theta}$ termed $\hat{\boldsymbol{\theta}}_{WLS}$ (pronounced theta-hat) by setting $\partial\epsilon/\partial\boldsymbol{\theta} = \boldsymbol{0}$ to obtain the normal equations

$$\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{WLS} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} \tag{115}$$

or $\boldsymbol{N}\hat{\boldsymbol{\theta}}_{WLS} = \boldsymbol{c}$ with $\boldsymbol{N} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ and $\boldsymbol{c} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y}$.

1.2.1 Parameter Estimates

If the symmetric matrix $\boldsymbol{N} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ is “well behaved”, i.e., it is full rank (equal to p) corresponding to linearly independent columns in $\boldsymbol{X}$ a formal solution is

$$\hat{\boldsymbol{\theta}}_{WLS} = (\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\,\boldsymbol{y} = \boldsymbol{N}^{-1}\boldsymbol{c}. \tag{116}$$

For reasons of numerical stability especially in situations with nearly linear dependencies between the columns of $\boldsymbol{X}$ (causing slight alterations to the observed values in $\boldsymbol{X}$ to lead to substantial changes in the estimated $\hat{\boldsymbol{\theta}}$; this problem is known as multicollinearity) the system of normal equations should not be solved by inverting $\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ but rather by means of SVD, QR or Cholesky decomposition, see Sections 1.1.6, 1.1.7 and 1.1.8.

When we apply regression analysis in other application areas we are often interested in predicting the response variable based on new data not used in the estimation of the parameters or the regression coefficients $\hat{\boldsymbol{\theta}}$. In land surveying and GNSS applications we are typically interested in $\hat{\boldsymbol{\theta}}$ and not on this predictive modelling.

[8] in Danish “spredningen på vægtenheden”

(In the linear case $\hat{\boldsymbol{\theta}}_{WLS}$ can be found in one go because $\boldsymbol{e}^T\boldsymbol{P}\boldsymbol{e}$ is quadratic in $\boldsymbol{\theta}$; unlike in the nonlinear case dealt with in Section 2 we don't need an initial value for $\boldsymbol{\theta}$ and an iterative procedure.)

The estimate for $\boldsymbol{y}$ termed $\hat{\boldsymbol{y}}$ (pronounced y-hat) is

$$\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\theta}}_{WLS} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} = \boldsymbol{H}\boldsymbol{y} = \boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{c} \tag{117}$$

where $\boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}$ is the so-called hat matrix since it transforms $\boldsymbol{y}$ into $\hat{\boldsymbol{y}}$. In geodesy (and land surveying) these equations are termed the fundamental equations. In WLS regression $\boldsymbol{H}$ is not symmetric, $\boldsymbol{H} \neq \boldsymbol{H}^T$. $\boldsymbol{H}$ is idempotent $\boldsymbol{H}\boldsymbol{H} = \boldsymbol{H}$. We also have $\boldsymbol{H}\boldsymbol{X} = \boldsymbol{X}$ and that the trace of $\boldsymbol{H}$, tr$\boldsymbol{H}$ = tr$(\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P})$ = tr$(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1})$ = tr$\boldsymbol{I}_p = p$. Also $\boldsymbol{P}\boldsymbol{H} = \boldsymbol{H}^T\boldsymbol{P} = \boldsymbol{H}^T\boldsymbol{P}\boldsymbol{H}$ which is symmetric.

The estimate of the error term $\boldsymbol{e}$ (also known as the residual) termed $\hat{e}$ (pronounced e-hat) is

$$\hat{e} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{H}\boldsymbol{y} = (\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y}. \tag{118}$$

In WLS regression $\boldsymbol{I} - \boldsymbol{H}$ is not symmetric, $\boldsymbol{I} - \boldsymbol{H} \neq (\boldsymbol{I} - \boldsymbol{H})^T$. $\boldsymbol{I} - \boldsymbol{H}$ is idempotent, $(\boldsymbol{I} - \boldsymbol{H})(\boldsymbol{I} - \boldsymbol{H}) = \boldsymbol{I} - \boldsymbol{H}$. We also have $(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{X} = \boldsymbol{0}$ and tr$(\boldsymbol{I} - \boldsymbol{H}) = n - p$. Also $\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{H}) = (\boldsymbol{I} - \boldsymbol{H})^T\boldsymbol{P} = (\boldsymbol{I} - \boldsymbol{H})^T\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{H})$ which is symmetric.

$\boldsymbol{X}$ and $\hat{e}$, and $\hat{\boldsymbol{y}}$ and $\hat{e}$ are orthogonal (with respect to $\boldsymbol{P}$): $\boldsymbol{X}^T\boldsymbol{P}\hat{e} = \boldsymbol{0}$ and $\hat{\boldsymbol{y}}^T\boldsymbol{P}\hat{e} = 0$. Geometrically this means that our analysis finds the orthogonal projection (with respect to $\boldsymbol{P}$) $\hat{\boldsymbol{y}}$ of $\boldsymbol{y}$ onto the hyperplane spanned by the linearly independent columns of $\boldsymbol{X}$. This gives the shortest distance between $\boldsymbol{y}$ and $\hat{\boldsymbol{y}}$ in the norm defined by $\boldsymbol{P}$.

Since the expectation of $\hat{\boldsymbol{\theta}}_{WLS}$ is

$$\begin{aligned}
\mathrm{E}\{\hat{\boldsymbol{\theta}}_{WLS}\} &= \mathrm{E}\{(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y}\} && (119)\\
&= (\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\mathrm{E}\{\boldsymbol{y}\} && (120)\\
&= (\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\mathrm{E}\{\boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e}\} && (121)\\
&= \boldsymbol{\theta}, && (122)
\end{aligned}$$

$\hat{\boldsymbol{\theta}}_{WLS}$ is unbiased or a central estimator.

1.2.2 Weight Assignment

In general we assign weights to observations so that the weight of an observation is proportional to the inverse expected (prior) variance of that observation, $p_i \propto 1/\sigma^2_{i,prior}$.

In traditional land surveying and GNSS we deal with observations of distances, directions and heights. In WLS we minimize half the weighted sum of squared residuals $\epsilon = 1/2\sum_{i=1}^{n} p_i e_i^2$. For this sum to make sense all terms must have the same unit. This can be obtained by demanding that $p_i e_i^2$ has no unit. This means that p_i has units of $1/e_i^2$ or $1/y_i^2$. If we consider the weight definition $\sigma_0^2 = p_1\sigma_1^2 = \cdots = p_i\sigma_i^2 = \cdots = p_n\sigma_n^2$ we see that σ_0^2 has no unit. Choosing $p_i = 1/\sigma^2_{i,prior}$ we obtain that $\sigma_0 = 1$ if measurements are carried out with the expected (prior) variances (and the regression model is correct). $\sigma_{i,prior}$ depends on the quality of the instruments applied and how measurements are performed. Below formulas for weights are given, see Jacobi (1977).

Distance Measurements Here we use

$$p_i = \frac{n}{s_G^2 + a^2 s_a^2} \tag{123}$$

where

- n is the number of observations,
- s_G is the combined expected standard deviation of the distance measuring instrument itself and on centering of the device,
- s_a is the expected distance dependent standard deviation of the distance measuring instrument, and
- a is the distance between the two points in question.

Directional Measurements Here we use

$$p_i = \frac{n}{n\frac{s_c^2}{a^2} + s_t^2} \tag{124}$$

where

- n is the number of observations,
- s_c is the expected standard deviation on centering of the device, and
- s_t is the expected standard deviation of one observed direction.

Levelling or Height Measurements Here we traditionally choose weights p_i equal to the number of measurements divided by the distance between the points in question measured in units of km, i.e., a weight of 1 is assigned to one measured height difference if that height difference is measured over a distance of 1 km. Since here in general $p_i \neq 1/\sigma^2_{i,prior}$ this choice of weights does not ensure $\sigma_0 = 1$. In this case the units for the weights are not those of the inverse prior variances so σ_0 is not unit-free, and also this tradition makes it impossible to carry out adjustment of combined height, direction and distance observations.

In conclusion we see that the weights for distances and directions change if the distance a between points change. The weights chosen for height measurements are generally not equal to the inverse of the expected (prior) variance of the observations. Therefore they do not lead to $\sigma_0 = 1$. Both distance and directional measurements lead to nonlinear least squares problems, see Section 2.

Example 7 (from Mærsk-Møller and Frederiksen, 1984, p. 74) From four points Q, A, B and C we have measured all possible pairwise height differences, see Figure 4. All measurements are carried out twice. Q has a known height $K_Q = 34.294$ m which is considered as fixed. We wish to determine the heights in points A, B and C by means of weighted least squares adjustment. These heights are called θ_1, θ_2 and θ_3 respectively. The mean of the two height measurements are (with the distance d_i between points in parentheses)

from Q to A 0.905 m (0.300 km),
from A to B 1.675 m (0.450 km),
from C to B 8.445 m (0.350 km),
from C to Q 5.864 m (0.300 km),
from Q to B 2.578 m (0.500 km), and
from C to A 6.765 m (0.450 km).

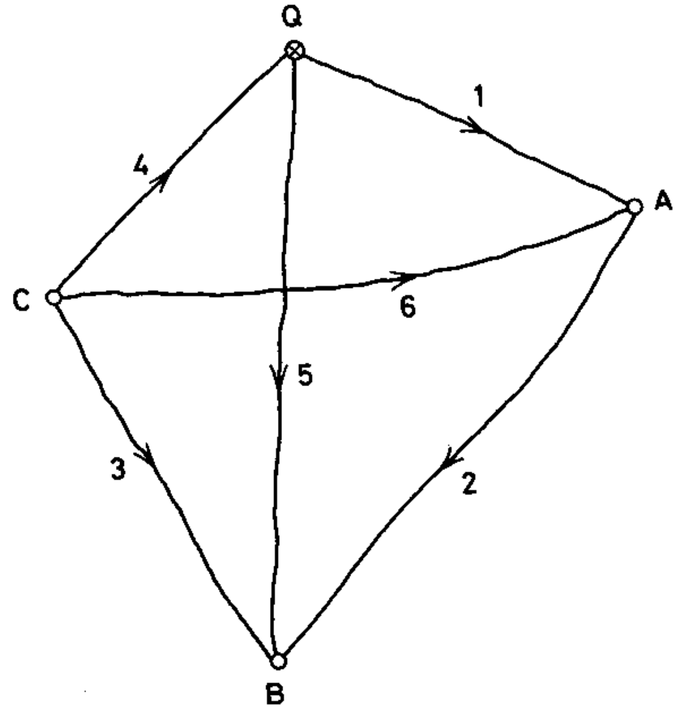

Figure 4: From four points Q, A, B and C we measure all possible pairwise height differences (from Mærsk-Møller and Frederiksen, 1984).

The weight for each observation is $p_i = 2/d_i$, see immediately above, resulting in (units are in km^{-1})

$$\boldsymbol{P} = \begin{bmatrix} 6.6667 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4.4444 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.7143 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6.6667 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.0000 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4.4444 \end{bmatrix}. \tag{125}$$

The six observation equations are

$$y_1 = \theta_1 - K_Q + e_1 \tag{126}$$
$$y_2 = \theta_2 - \theta_1 + e_2 \tag{127}$$
$$y_3 = \theta_2 - \theta_3 + e_3 \tag{128}$$
$$y_4 = K_Q - \theta_3 + e_4 \tag{129}$$
$$y_5 = \theta_2 - K_Q + e_5 \tag{130}$$
$$y_6 = \theta_1 - \theta_3 + e_6. \tag{131}$$

In matrix form we get (units are m)

$$\begin{bmatrix} 0.905 \\ 1.675 \\ 8.445 \\ 5.864 \\ 2.578 \\ 6.765 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} + \begin{bmatrix} -34.294 \\ 0.000 \\ 0.000 \\ 34.294 \\ -34.294 \\ 0.000 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix} \tag{132}$$

or (with a slight misuse of notation since we reuse the θ_is and the e_is; this is $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta} + \boldsymbol{e}$; units are mm)

$$\begin{bmatrix} 35,199 \\ 1,675 \\ 8,445 \\ -28,430 \\ 36,872 \\ 6,765 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix}. \tag{133}$$

The normal equations are (this is $\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\hat{\boldsymbol{\theta}} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y}$; units are mm)

$$\begin{bmatrix} 15.5556 & -4.4444 & -4.4444 \\ -4.4444 & 14.1587 & -5.7143 \\ -4.4444 & -5.7143 & 16.8254 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 257,282.22 \\ 202,189.59 \\ 111,209.52 \end{bmatrix}. \tag{134}$$

The solution is $\hat{\boldsymbol{\theta}} = [35,197.8\ 36,873.6\ 28,430.3]^T$ mm, see Matlab code in page 23. [end of example]

1.2.3 Dispersion and Significance of Estimates

Dispersion or variance-covariance matrices for $\boldsymbol{y}$, $\hat{\boldsymbol{\theta}}_{WLS}$, $\hat{\boldsymbol{y}}$ and $\hat{\boldsymbol{e}}$ are

$$\begin{aligned} \mathrm{D}\{\boldsymbol{y}\} &= \sigma_0^2\boldsymbol{P}^{-1} & (135)\\ \mathrm{D}\{\hat{\boldsymbol{\theta}}_{WLS}\} &= \sigma_0^2(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1} = \sigma_0^2\boldsymbol{N}^{-1} & (136)\\ \mathrm{D}\{\hat{\boldsymbol{y}}\} &= \sigma_0^2\boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{X}^T = \sigma_0^2\boldsymbol{H}\boldsymbol{P}^{-1} & (137)\\ \mathrm{D}\{\hat{\boldsymbol{e}}\} &= \sigma_0^2(\boldsymbol{P}^{-1} - \boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{X}^T) & (138)\\ &= \sigma_0^2(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{P}^{-1} = \mathrm{D}\{\boldsymbol{y}\} - \mathrm{D}\{\hat{\boldsymbol{y}}\}. & (139) \end{aligned}$$

We see that since the dispersion of $\hat{\boldsymbol{y}}$ is not proportional to the hat matrix, an alternative measure of leverage in this case is $\boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{X}^T$ (although with units as opposed to the elements of the hat matrix which are unit free). This alternative measure may be useful only if measurements have the same units and are on the same scale.

For the weighted sum of squared errors (SSE, also called RSS for the residual sum of squares) we get

$$\begin{aligned} \hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}} &= \boldsymbol{y}^T(\boldsymbol{I} - \boldsymbol{H})^T\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y} & (140)\\ &= \boldsymbol{y}^T(\boldsymbol{I} - \boldsymbol{H})^T(\boldsymbol{P} - \boldsymbol{P}\boldsymbol{H})\boldsymbol{y} & (141)\\ &= \boldsymbol{y}^T(\boldsymbol{P} - \boldsymbol{P}\boldsymbol{H} - \boldsymbol{H}^T\boldsymbol{P} + \boldsymbol{H}^T\boldsymbol{P}\boldsymbol{H})\boldsymbol{y} & (142)\\ &= \boldsymbol{y}^T\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{y} & (143) \end{aligned}$$

with expectation $\mathrm{E}\{\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}\} = \sigma_0^2(n-p)$. The mean squared error MSE is

$$\hat{\sigma}_0^2 = \hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}/(n-p) \tag{144}$$

and the root mean squared error RMSE is $\hat{\sigma}_0$ also known as s_0. $\hat{\sigma}_0 = s_0$ has no unit. For well performed measurements (with no outliers or gross errors), a good model, and properly chosen weights (see Section 1.2.2), $s_0 \simeq 1$. This is due to the fact that assuming that the e_is with variance σ_0^2/p_i are independent and normally distributed, $\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}$ (with well chosen p_i, see Section 1.2.2) follows a χ^2 distribution with $n-p$ degrees of freedom which has expectation $n-p$. Therefore $\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}/(n-p)$ has expectation 1 and its square root is approximately 1.

What if s_0 is larger than 1? How much larger than 1 is too large? If we assume that the e_is are independent and follow a normal distribution, $\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}} = (n-p)s_0^2$ follows a χ^2 distribution with $n-p$ degrees of freedom. If the probability of finding $(n-p)s_0^2$ larger than the observed value is much smaller than the traditionally used 0.05 (5%) or 0.01 (1%), then s_0 is too large.

The square roots of the diagonal elements of the dispersion matrices in Equations 135, 136, 137 and 138 are the standard errors of the quantities in question. For example, the standard error of $\hat{\theta}_i$ denoted $\hat{\sigma}_{\theta_i}$ is the square root of the ith diagonal element of $\sigma_0^2(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}$.

As in the OLS case we can define standardized residuals

$$e_i' = \frac{\hat{e}_i}{\hat{\sigma}_0}\sqrt{\frac{p_i}{1-H_{ii}}} \tag{145}$$

with unit variance.

Example 8 (continuing Example 7) The hat matrix is

$$\boldsymbol{H} = \begin{bmatrix} 0.5807 & -0.1985 & 0.0287 & -0.2495 & 0.1698 & 0.2208 \\ -0.2977 & 0.4655 & 0.2941 & -0.0574 & 0.2403 & -0.2367 \\ 0.0335 & 0.2288 & 0.5452 & 0.2595 & 0.2260 & 0.1953 \\ -0.2495 & -0.0383 & 0.2224 & 0.5664 & -0.1841 & 0.2112 \\ 0.2830 & 0.2670 & 0.3228 & -0.3069 & 0.4101 & -0.0159 \\ 0.3312 & -0.2367 & 0.2511 & 0.3169 & -0.0143 & 0.4320 \end{bmatrix} \tag{146}$$

and $p/n = 3/6 = 0.5$, no diagonal element is higher than two (or three) times 0.5. The diagonal of

$$\boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{X}^T = \begin{bmatrix} 0.0871 & -0.0447 & 0.0050 & -0.0374 & 0.0424 & 0.0497 \\ -0.0447 & 0.1047 & 0.0515 & -0.0086 & 0.0601 & -0.0533 \\ 0.0050 & 0.0515 & 0.0954 & 0.0389 & 0.0565 & 0.0439 \\ -0.0374 & -0.0086 & 0.0389 & 0.0850 & -0.0460 & 0.0475 \\ 0.0424 & 0.0601 & 0.0565 & -0.0460 & 0.1025 & -0.0036 \\ 0.0497 & -0.0533 & 0.0439 & 0.0475 & -0.0036 & 0.0972 \end{bmatrix} \mathrm{mm}^2 \tag{147}$$

is an alternative measure of leverage and no (diagonal) element is larger than two or three times the average, i.e, no observations have high leverages. In this case where all measurements have the same units and are on the same scale both measures of leverage appear sensible. (See also Example 10 where this is not the case.) Checking the diagonal of $\boldsymbol{X}\boldsymbol{N}^{-1}\boldsymbol{X}^T$ of course corresponds to checking the variances (or standard deviations) of the predicted observations $\hat{\boldsymbol{y}}$.

The estimated residuals are $\hat{\boldsymbol{e}} = [1.1941\ -0.7605\ 1.6879\ 0.2543\ -1.5664\ -2.5516]^T$ mm. Therefore the RMSE or $\hat{\sigma}_0 = s_0 = \sqrt{\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}/3}$ mm/km$^{1/2}$ = 4.7448 mm/km$^{1/2}$. The inverse of $\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ is

$$\begin{bmatrix} 15.556 & -4.4444 & -4.4444 \\ -4.4444 & 14.159 & -5.7143 \\ -4.4444 & -5.7143 & 16.825 \end{bmatrix}^{-1} = \begin{bmatrix} 0.087106 & 0.042447 & 0.037425 \\ 0.042447 & 0.10253 & 0.046034 \\ 0.037425 & 0.046034 & 0.084954 \end{bmatrix}. \tag{148}$$

This gives standard deviations for $\boldsymbol{\theta}$, $\hat{\sigma}_\theta = [1.40\ 1.52\ 1.38]^T$ mm.

Although the weighting scheme for levelling is not designed to give $s_0 = 1$ (with no unit) we look into the magnitude of s_0 for illustration. s_0 is larger than 1. Had the weighting scheme been designed to obtain $s_0 = 1$ (with no unit) would $s_0 = 4.7448$ be too large? If the e_is are independent and follow a normal distribution, $\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}} = (n-p)s_0^2$ follows a χ^2 distribution with three degrees of freedom. The probability of finding $(n-p)s_0^2$ larger than the observed $3 \times 4.7448^2 = 67.5382$ is smaller than 10^{-13} which is much smaller than the traditionally used 0.05 or 0.01. So s_0 is too large. Judged from the residuals, the standard deviations and the

t-test statistics (see Example 9) the fit to the model is excellent. Again for illustration: had the weights been one tenth of the values used above, s_0 would be $4.7448/\sqrt{10} = 1.5004$, again larger than 1. The probability of finding $(n-p)s_0^2 > 3 \times 1.5004^2 = 6.7538$ is 0.0802. Therefore this value of s_0 would be suitably small. [end of example]

If we assume that the e_is are independent and follow a normal distribution $\hat{\boldsymbol{\theta}}_{WLS}$ follows a multivariate normal distribution with mean $\boldsymbol{\theta}$ and dispersion $\sigma_0^2(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}$. Assuming that $\hat{\theta}_i = c_i$ where c_i is a constant it can be shown that the ratio

$$z_i = \frac{\hat{\theta}_i - c_i}{\hat{\sigma}_{\theta_i}} \tag{149}$$

follows a t distribution with $n-p$ degrees of freedom. This can be used to test whether $\hat{\theta}_i - c_i$ is significantly different from 0. If for example z_i with $c_i = 0$ has a small absolute value then $\hat{\theta}_i$ is not significantly different from 0 and x_i should be removed from the model.

Example 9 (continuing Example 8) The t-test statistics z_i with $c_i = 0$ are $[25,135\ 24,270\ 20,558]^T$ which are all extremely large compared to 95% or 99% percentiles in a two-sided t-test with three degrees of freedom, 3.182 and 5.841 respectively. To double precision the probabilities of finding larger values of $|z_i|$ are $[0\ 0\ 0]^T$. All parameter estimates are significantly different from zero. [end of example]

Matlab code for Examples 7 to 9

```
% (C) Copyright 2003
% Allan Aasbjerg Nielsen
% aa@imm.dtu.dk, www.imm.dtu.dk/˜aa

Kq = 34.294;
X = [1 0 0;-1 1 0;0 1 -1;0 0 -1;0 1 0;1 0 -1];
[n p] = size(X);
%number of degrees of freedom
f = n-p;
dist = [0.30 0.45 0.35 0.30 0.50 0.45];
P = diag(2./dist); % units [kmˆ(-1)]
%P = 0.1*P; % This gives a better s0
%OLS
%P = eye(size(X,1));
y = [.905 1.675 8.445 5.864 2.578 6.765]';

%units are mm
y = 1000*y;
Kq = 1000*Kq;

cst = Kq.*[1 0 0 -1 1 0]';
y = y+cst;
%OLS by "\" operator: mldivide
%thetahat = X'*X\(X'*y)
N = X'*P;
c = N*y;
N = N*X;
%WLS
thetahat = N\c;
yhat = X*thetahat;
ehat = y-yhat;
yhat = yhat-cst;
%MSE
SSE = ehat'*P*ehat;
s02 = SSE/f;
%RMSE
s0 = sqrt(s02);

%Variance/covariance matrix of the observations, y
Dy = s02.*inv(P);
%Standard deviations
stdy = sqrt(diag(Dy));

%Variance/covariance matrix of the adjusted elements, thetahat
Ninv = inv(N);
Dthetahat = s02.*Ninv;
%Standard deviations
stdthetahat = sqrt(diag(Dthetahat));
```

```
%Variance/covariance matrix of the adjusted observations, yhat
Dyhat = s02.*X*Ninv*X';
%Standard deviations
stdyhat = sqrt(diag(Dyhat));

%Variance/covariance matrix of the adjusted residuals, ehat
Dehat = Dy-Dyhat;
%Standard deviations
stdehat = sqrt(diag(Dehat));

%Correlations between adjusted elements, thetahat
aux = diag(1./stdthetahat);
corthetahat = aux*Dthetahat*aux;

% tests

% t-values and probabilities of finding larger |t|
% pt should be smaller than, say, (5% or) 1%
t = thetahat./stdthetahat;
pt = betainc(f./(f+t.^2),0.5*f,0.5);

% probability of finding larger s02
% should be greater than, say, 5% (or 1%)
pchi2 = 1-gammainc(0.5*SSE,0.5*f);
```

Probabilities in the χ^2 distribution are calculated by means of the incomplete gamma function evaluated in Matlab by the `gammainc` function.

A Trick to Obtain $\sqrt{\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}}$ with the Cholesky Decomposition

$\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X} = \boldsymbol{C}\boldsymbol{C}^T$, $\boldsymbol{C}$ $p \times p$ lower triangular

$$\boldsymbol{C}\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{WLS} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} \tag{150}$$

$$\boldsymbol{C}(\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{WLS}) = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} \tag{151}$$

so $\boldsymbol{C}\boldsymbol{z} = \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y}$ with $\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{WLS} = \boldsymbol{z}$. Expand $p \times p$ $\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}$ with one more row and column to $(p+1) \times (p+1)$

$$\tilde{\boldsymbol{C}}\tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X} & \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} \\ (\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y})^T & \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} \end{bmatrix}. \tag{152}$$

With

$$\tilde{\boldsymbol{C}} = \begin{bmatrix} \boldsymbol{C} & \boldsymbol{0} \\ \boldsymbol{z}^T & s \end{bmatrix} \text{ and } \tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{C}^T & \boldsymbol{z} \\ \boldsymbol{0}^T & s \end{bmatrix} \tag{153}$$

we get

$$\tilde{\boldsymbol{C}}\tilde{\boldsymbol{C}}^T = \begin{bmatrix} \boldsymbol{C}\boldsymbol{C}^T & \boldsymbol{C}\boldsymbol{z} \\ \boldsymbol{z}^T\boldsymbol{C}^T & \boldsymbol{z}^T\boldsymbol{z} + s^2 \end{bmatrix}. \tag{154}$$

We see that

$$\begin{aligned}
s^2 &= \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \boldsymbol{z}^T\boldsymbol{z} && (155)\\
&= \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \hat{\boldsymbol{\theta}}^T_{WLS}\boldsymbol{C}\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{WLS} && (156)\\
&= \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \hat{\boldsymbol{\theta}}^T_{WLS}\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} && (157)\\
&= \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{WLS} && (158)\\
&= \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} && (159)\\
&= \boldsymbol{y}^T\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{P})\boldsymbol{y} && (160)\\
&= \hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}. && (161)
\end{aligned}$$

Hence, after Cholesky decomposition of the expanded matrix, the lower right element of $\tilde{\boldsymbol{C}}$ is $\sqrt{\hat{\boldsymbol{e}}^T\boldsymbol{P}\hat{\boldsymbol{e}}}$. The last column in $\tilde{\boldsymbol{C}}^T$ (skipping s in the last row) is $\boldsymbol{C}^T\hat{\boldsymbol{\theta}}_{WLS}$, hence $\hat{\boldsymbol{\theta}}_{WLS}$ can be found by back-substitution.

1.2.4 WLS as OLS

The WLS problem can be turned into an OLS problem by replacing $\boldsymbol{X}$ by $\tilde{\boldsymbol{X}} = \boldsymbol{P}^{1/2}\boldsymbol{X}$ and $\boldsymbol{y}$ by $\tilde{\boldsymbol{y}} = \boldsymbol{P}^{1/2}\boldsymbol{y}$ with $\boldsymbol{P}^{1/2} = \text{diag}[\sqrt{p_1}, \ldots, \sqrt{p_n}]$ to get the OLS normal equations

$$\begin{aligned}
\boldsymbol{X}^T\boldsymbol{P}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{WLS} &= \boldsymbol{X}^T\boldsymbol{P}\boldsymbol{y} & (162)\\
(\boldsymbol{P}^{1/2}\boldsymbol{X})^T(\boldsymbol{P}^{1/2}\boldsymbol{X})\hat{\boldsymbol{\theta}}_{WLS} &= (\boldsymbol{P}^{1/2}\boldsymbol{X})^T(\boldsymbol{P}^{1/2}\boldsymbol{y}) & (163)\\
\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}}\hat{\boldsymbol{\theta}}_{WLS} &= \tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{y}}. & (164)
\end{aligned}$$

1.3 General Least Squares, GLS

In GLS the residuals may be correlated and we assume that $\text{D}\{\boldsymbol{y}\} = \text{D}\{\boldsymbol{e}\} = \sigma_0^2\Sigma$. So $\sigma_0^2\Sigma$ is the dispersion or variance-covariance matrix of the residuals possibly with off-diagonal elements. This may be the case for instance when we work on differenced data and not directly on observed data. We minimize the objective function $\epsilon = \boldsymbol{e}^T\Sigma^{-1}\boldsymbol{e}/2$

$$\begin{aligned}
\epsilon &= 1/2(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta})^T\Sigma^{-1}(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}) & (165)\\
&= 1/2(\boldsymbol{y}^T\Sigma^{-1}\boldsymbol{y} - \boldsymbol{y}^T\Sigma^{-1}\boldsymbol{X}\boldsymbol{\theta} - \boldsymbol{\theta}^T\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\boldsymbol{\theta}) & (166)\\
&= 1/2(\boldsymbol{y}^T\Sigma^{-1}\boldsymbol{y} - 2\boldsymbol{\theta}^T\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\boldsymbol{\theta}). & (167)
\end{aligned}$$

Just as in the WLS case we obtain the normal equations

$$\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{GLS} = \boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y}. \tag{168}$$

If the symmetric matrix $\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}$ is "well behaved", i.e., it is full rank (equal to p) corresponding to linearly independent columns in $\boldsymbol{X}$ a formal solution is

$$\hat{\boldsymbol{\theta}}_{GLS} = (\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T\Sigma^{-1}\,\boldsymbol{y} \tag{169}$$

with dispersion

$$\text{D}\{\hat{\boldsymbol{\theta}}_{GLS}\} = \sigma_0^2(\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X})^{-1}. \tag{170}$$

As with OLS and WLS

$$\hat{\boldsymbol{e}} = \boldsymbol{y} - \hat{\boldsymbol{y}} = \boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\theta}}_{GLS}. \tag{171}$$

For the dispersion of $\hat{\boldsymbol{y}}$ we get

$$\text{D}\{\hat{\boldsymbol{y}}\} = \sigma_0^2\boldsymbol{X}(\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X})^{-1}\boldsymbol{X}^T. \tag{172}$$

The mean squared error MSE is

$$\hat{\sigma}_0^2 = \hat{\boldsymbol{e}}^T\Sigma^{-1}\hat{\boldsymbol{e}}/(n-p) \tag{173}$$

and the root mean squared error RMSE is $\hat{\sigma}_0$ also known as s_0.

The GLS problem can be turned into an OLS problem by means of the Cholesky decomposition of $\Sigma = \boldsymbol{C}\boldsymbol{C}^T$ (or of Σ^{-1})

$$\begin{aligned}
\boldsymbol{X}^T\Sigma^{-1}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{GLS} &= \boldsymbol{X}^T\Sigma^{-1}\boldsymbol{y} & (174)\\
\boldsymbol{X}^T\boldsymbol{C}^{-T}\boldsymbol{C}^{-1}\boldsymbol{X}\hat{\boldsymbol{\theta}}_{GLS} &= \boldsymbol{X}^T\boldsymbol{C}^{-T}\boldsymbol{C}^{-1}\boldsymbol{y} & (175)\\
(\boldsymbol{C}^{-1}\boldsymbol{X})^T(\boldsymbol{C}^{-1}\boldsymbol{X})\hat{\boldsymbol{\theta}}_{GLS} &= (\boldsymbol{C}^{-1}\boldsymbol{X})^T(\boldsymbol{C}^{-1}\boldsymbol{y}) & (176)\\
\tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{X}}\hat{\boldsymbol{\theta}}_{GLS} &= \tilde{\boldsymbol{X}}^T\tilde{\boldsymbol{y}}, & (177)
\end{aligned}$$

i.e., replace $\boldsymbol{X}$ by $\tilde{\boldsymbol{X}} = \boldsymbol{C}^{-1}\boldsymbol{X}$ and $\boldsymbol{y}$ by $\tilde{\boldsymbol{y}} = \boldsymbol{C}^{-1}\boldsymbol{y}$.

1.3.1 Regularization

In the so-called regularized or penalized case we penalize some characteristic of $\boldsymbol{\theta}$, for example size, by introducing an extra term into Equation 165, namely $\lambda\boldsymbol{\theta}^T\boldsymbol{\Omega}\boldsymbol{\theta}$ where $\boldsymbol{\Omega}$ describes some characteristic of $\boldsymbol{\theta}$ and the small positive scalar λ determines the amount of regularization. If we wish to penalize large θ_i, i.e., we wish to penalize size, $\boldsymbol{\Omega}$ is the unit matrix. In the regularized case the normal equations become

$$(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}+\lambda\boldsymbol{\Omega})\tilde{\boldsymbol{\theta}}_{GLS}=\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{y}, \tag{178}$$

the dispersion of $\tilde{\boldsymbol{\theta}}_{GLS}$ becomes

$$\mathrm{D}\{\tilde{\boldsymbol{\theta}}_{GLS}\}=\sigma_0^2(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}+\lambda\boldsymbol{\Omega})^{-1}\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}+\lambda\boldsymbol{\Omega})^{-1} \tag{179}$$

leading to this dispersion of $\hat{\boldsymbol{y}}$

$$\mathrm{D}\{\hat{\boldsymbol{y}}\}=\sigma_0^2\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}+\lambda\boldsymbol{\Omega})^{-1}\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{X}+\lambda\boldsymbol{\Omega})^{-1}\boldsymbol{X}^T. \tag{180}$$

2 Nonlinear Least Squares

Consider y as a general, nonlinear function of the θ_js where f can subsume a constant term if present

$$y_i=f_i(\theta_1,\dots,\theta_p)+e_i,\ i=1,\dots,n. \tag{181}$$

In the traditional land surveying notation of Mærsk-Møller and Frederiksen (1984) we have ($y_i\sim\ell_i,\ f_i\sim F_i,\ \theta_j\sim x_j$, and $e_i\sim v_i$)

$$\ell_i=F_i(x_1,\dots,x_p)+v_i,\ i=1,\dots,n. \tag{182}$$

(Mærsk-Møller and Frederiksen (1984) use $-v_i$; whether we use $+v_i$ or $-v_i$ is irrelevant for LS methods.) Several methods are available to solve this problem, see Sections 2.2.1, 2.2.2, 2.2.3 and 2.2.4. Here we use a linearization method.

If we have one parameter x only we get (we omit the observation index i)

$$\ell=F(x)+v. \tag{183}$$

In geodesy and land surveying the parameters are often called elements. We perform a Taylor expansion of F around a chosen initial value x^*[9]

$$\ell=F(x^*)+F'(x^*)(x-x^*)+\frac{1}{2!}F''(x^*)(x-x^*)^2+\frac{1}{3!}F'''(x^*)(x-x^*)^3+\cdots+v \tag{184}$$

and retain up till the first order term only (i.e., we linearize F near x^* to approximate v^2 to a quadratic near x^*; a single prime $'$ denotes the first order derivative, two primes $''$ denote the second order derivative etc.)

$$\ell\simeq F(x^*)+F'(x^*)(x-x^*)+v. \tag{185}$$

Geometrically speaking we work on the tangent of $F(x)$ at x^*.

If we have p parameters or elements $\boldsymbol{x}=[x_1,\dots,x_p]^T$ we get

$$\ell=F(x_1,\dots,x_p)+v=F(\boldsymbol{x})+v \tag{186}$$

[9] in Danish “foreløbig værdi” or “foreløbigt element”

and from a Taylor expansion we retain the first order terms only

$$\ell \simeq F(x_1^*,\ldots,x_p^*) + \left.\frac{\partial F}{\partial x_1}\right|_{x_1=x_1^*}(x_1-x_1^*) + \cdots + \left.\frac{\partial F}{\partial x_p}\right|_{x_p=x_p^*}(x_p-x_p^*) + v \tag{187}$$

or

$$\ell \simeq F(\boldsymbol{x}^*) + [\nabla F(\boldsymbol{x}^*)]^T(\boldsymbol{x}-\boldsymbol{x}^*) + v \tag{188}$$

where $\nabla F(\boldsymbol{x}^*)$ is the gradient of F, $[\nabla F(\boldsymbol{x}^*)]^T = [\partial F/\partial x_1 \ \ldots \ \partial F/\partial x_p]_{\boldsymbol{x}=\boldsymbol{x}^*}$, evaluated at $\boldsymbol{x} = \boldsymbol{x}^* = [x_1^*,\ldots,x_p^*]^T$. Geometrically speaking we work in the tangent hyperplane of $F(\boldsymbol{x})$ at $\boldsymbol{x}^*$.

Write all n equations in vector notation

$$\begin{bmatrix} \ell_1 \\ \ell_2 \\ \vdots \\ \ell_n \end{bmatrix} = \begin{bmatrix} F_1(\boldsymbol{x}) \\ F_2(\boldsymbol{x}) \\ \vdots \\ F_n(\boldsymbol{x}) \end{bmatrix} + \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} \tag{189}$$

or

$$\boldsymbol{\ell} = \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{v} \tag{190}$$

and get

$$\boldsymbol{\ell} \simeq \boldsymbol{F}(\boldsymbol{x}^*) + \boldsymbol{A}(\boldsymbol{x}-\boldsymbol{x}^*) + \boldsymbol{v} \tag{191}$$

where the $n \times p$ derivative matrix $\boldsymbol{A}$ is

$$\boldsymbol{A} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} = \left[\frac{\partial \boldsymbol{F}}{\partial x_1} \ \cdots \ \frac{\partial \boldsymbol{F}}{\partial x_p}\right] = \begin{bmatrix} \frac{\partial F_1}{\partial x_1} & \cdots & \frac{\partial F_1}{\partial x_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial x_1} & \cdots & \frac{\partial F_n}{\partial x_p} \end{bmatrix} \tag{192}$$

with all $A_{ij} = \partial F_i/\partial x_j$ evaluated at $x_j = x_j^*$. Therefore we get (here we use "$=$" instead of the correct "$\simeq$")

$$\boldsymbol{k} = \boldsymbol{A\Delta} + \boldsymbol{v} \tag{193}$$

where $\boldsymbol{k} = \boldsymbol{\ell} - \boldsymbol{F}(\boldsymbol{x}^*)$ and $\boldsymbol{\Delta} = \boldsymbol{x} - \boldsymbol{x}^*$ (Mærsk-Møller and Frederiksen (1984) use $\boldsymbol{k} = \boldsymbol{F}(\boldsymbol{x}^*) - \boldsymbol{\ell}$). $\hat{\boldsymbol{\ell}} = \boldsymbol{F}(\hat{\boldsymbol{x}})$ are termed the fundamental equations in geodesy and land surveying. Equations 190 and 193 are termed the observation equations. Equation 193 is a linearized version.

2.1 Nonlinear WLS by Linearization

If we compare $\boldsymbol{k} = \boldsymbol{A\Delta} + \boldsymbol{v}$ in Equation 193 with the linear expression $\boldsymbol{y} = \boldsymbol{X\theta} + \boldsymbol{e}$ in Equation 34 and the normal equations for the linear WLS problem in Equation 115, we get the normal equations for the WLS estimate $\hat{\boldsymbol{\Delta}}$ of the increment $\boldsymbol{\Delta}$

$$\boldsymbol{A}^T\boldsymbol{PA}\hat{\boldsymbol{\Delta}} = \boldsymbol{A}^T\boldsymbol{Pk} \tag{194}$$

or $\boldsymbol{N}\hat{\boldsymbol{\Delta}} = \boldsymbol{c}$ with $\boldsymbol{N} = \boldsymbol{A}^T\boldsymbol{PA}$ and $\boldsymbol{c} = \boldsymbol{A}^T\boldsymbol{Pk}$ (Mærsk-Møller and Frederiksen (1984) use $-\boldsymbol{k}$ and therefore also $-\boldsymbol{c}$).

2.1.1 Parameter Estimates

If the symmetric matrix $\boldsymbol{N} = \boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}$ is "well behaved", i.e., it is full rank (equal to p) corresponding to linearly independent columns in $\boldsymbol{A}$ a formal solution is

$$\hat{\boldsymbol{\Delta}} = (\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{P}\,\boldsymbol{k} = \boldsymbol{N}^{-1}\boldsymbol{c}. \tag{195}$$

For reasons of numerical stability especially in situations with nearly linear dependencies between the columns of $\boldsymbol{A}$ (causing slight alterations to the values in $\boldsymbol{A}$ to lead to substantial changes in the estimated $\hat{\boldsymbol{\Delta}}$; this problem is known as multicollinearity) the system of normal equations should not be solved by inverting $\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}$ but rather by means of SVD, QR or Cholesky decomposition, see Sections 1.1.6, 1.1.7 and 1.1.8.

When we apply regression analysis in other application areas we are often interested in predicting the response variable based on new data not used in the estimation of the parameters or the regression coefficients $\hat{\boldsymbol{x}}$. In land surveying and GNSS applications we are typically interested in $\hat{\boldsymbol{x}}$ and not on this predictive modelling.

2.1.2 Iterative Solution

To find the solution we update $\boldsymbol{x}^*$ to $\boldsymbol{x}^* + \hat{\boldsymbol{\Delta}}$ and go again. For how long do we "go again" or iterate? Until the elements in $\hat{\boldsymbol{\Delta}}$ become small, or based on a consideration in terms of the sum of weighted squared residuals

$$\begin{aligned}
\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}} &= (\boldsymbol{k} - \boldsymbol{A}\hat{\boldsymbol{\Delta}})^T\boldsymbol{P}(\boldsymbol{k} - \boldsymbol{A}\hat{\boldsymbol{\Delta}}) &(196)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{A}\hat{\boldsymbol{\Delta}} - \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} + \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}\hat{\boldsymbol{\Delta}} &(197)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{A}\hat{\boldsymbol{\Delta}} - \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} + \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A}(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} &(198)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{A}\hat{\boldsymbol{\Delta}} - \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} + \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} &(199)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{A}\hat{\boldsymbol{\Delta}} &(200)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \hat{\boldsymbol{\Delta}}^T\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{k} &(201)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \hat{\boldsymbol{\Delta}}^T\boldsymbol{c} &(202)\\
&= \boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k} - \boldsymbol{c}^T\boldsymbol{N}^{-1}\boldsymbol{c}. &(203)
\end{aligned}$$

Hence

$$\frac{\boldsymbol{k}^T\boldsymbol{P}\boldsymbol{k}}{\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}} = 1 + \frac{\boldsymbol{c}^T\boldsymbol{N}^{-1}\boldsymbol{c}}{\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}} \geq 1. \tag{204}$$

Therefore we iterate until the ratio of the two quadratic forms on the right hand side is small compared to 1.

The method described here is identical to the Gauss-Newton method sketched in Section 2.2.3 with $-\boldsymbol{A}$ as the Jacobian.

2.1.3 Dispersion and Significance of Estimates

When iterations are over and we have a solution we find dispersion or variance-covariance matrices for $\boldsymbol{\ell}$, $\hat{\boldsymbol{x}}$, $\hat{\boldsymbol{\ell}}$ and $\hat{\boldsymbol{v}}$ (again by analogy with the linear WLS case; the $\boldsymbol{Q}$s are (nearly) Mærsk-Møller and Frederiksen (1984) notation, and again we use "=" instead of the correct "$\simeq$")

$$\begin{aligned}
\boldsymbol{Q}_\ell &= \mathrm{D}\{\boldsymbol{\ell}\} &&= \sigma_0^2\boldsymbol{P}^{-1} &(205)\\
\boldsymbol{Q}_{\hat{x}} &= \mathrm{D}\{\hat{\boldsymbol{x}}_{WLS}\} &&= \sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1} = \sigma_0^2\boldsymbol{N}^{-1} &(206)\\
\boldsymbol{Q}_{\hat{\ell}} &= \mathrm{D}\{\hat{\boldsymbol{\ell}}\} &&= \sigma_0^2\boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T &(207)\\
\boldsymbol{Q}_{\hat{v}} &= \mathrm{D}\{\hat{\boldsymbol{v}}\} &&= \sigma_0^2(\boldsymbol{P}^{-1} - \boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T) = \mathrm{D}\{\boldsymbol{\ell}\} - \mathrm{D}\{\hat{\boldsymbol{\ell}}\} = \boldsymbol{Q}_\ell - \boldsymbol{Q}_{\hat{\ell}}. &(208)
\end{aligned}$$

For the weighted sum of squared errors (SSE, also called RSS for the residual sum of squares) we get

$$\hat{\boldsymbol{v}}^T \boldsymbol{P} \hat{\boldsymbol{v}} = \boldsymbol{k}^T \boldsymbol{P}(\boldsymbol{I} - \boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T\boldsymbol{P})\boldsymbol{k} = \boldsymbol{k}^T\boldsymbol{P}(\boldsymbol{I} - \boldsymbol{H})\boldsymbol{k} \tag{209}$$

with expectation $\mathrm{E}\{\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}\} = \sigma_0^2(n-p)$. $\boldsymbol{H} = \boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T\boldsymbol{P}$. The mean squared error MSE is

$$\hat{\sigma}_0^2 = \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}/(n-p) \tag{210}$$

and the root mean squared error RMSE is $\hat{\sigma}_0$ also known as s_0. $\hat{\sigma}_0 = s_0$ has no unit. For well performed measurements (with no outliers or gross errors), a good model, and properly chosen weights (see Section 1.2.2), $s_0 \simeq 1$. This is due to the fact that assuming that the v_is with variance σ_0^2/p_i are independent and normally distributed $\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}$ (with well chosen p_i, see Section 1.2.2) follows a χ^2 distribution with $n-p$ degrees of freedom which has expectation $n-p$. Therefore $\hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}/(n-p)$ has expectation 1 and its square root is approximately 1.

The square roots of the diagonal elements of the dispersion matrices in Equations 205, 206, 207 and 208 are the standard errors of the quantities in question. For example, the standard error of $\hat{x}_i$ denoted $\hat{\sigma}_{x_i}$ is the square root of the ith diagonal element of $\sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}$.

The remarks on 1) the distribution and significance of $\hat{\boldsymbol{\theta}} \sim \hat{\boldsymbol{x}}$ in Section 1.2.3, and 2) on influence and leverage in Section 1.1.5, are valid here also.

$\boldsymbol{A}$ and $\hat{\boldsymbol{v}}$, and $\hat{\boldsymbol{k}}$ and $\hat{\boldsymbol{v}}$ are orthogonal (with respect to $\boldsymbol{P}$): $\boldsymbol{A}^T\boldsymbol{P}\hat{\boldsymbol{v}} = \boldsymbol{0}$ and $\hat{\boldsymbol{k}}^T\boldsymbol{P}\hat{\boldsymbol{v}} = 0$. Geometrically this means that our analysis finds the orthogonal projection (with respect to $\boldsymbol{P}$) $\hat{\boldsymbol{k}}$ of $\boldsymbol{k}$ onto the hyperplane spanned by the linearly independent columns of $\boldsymbol{A}$. This gives the shortest distance between $\boldsymbol{k}$ and $\hat{\boldsymbol{k}}$ in the norm defined by $\boldsymbol{P}$.

2.1.4 Confidence Ellipsoids

We already described the quality of the estimates in $\hat{\boldsymbol{x}}$ by means of their standard deviations, i.e., the square roots of the diagonal elements of $\mathrm{D}\{\hat{\boldsymbol{x}}\} = \boldsymbol{Q}_{\hat{x}}$. Another description which allows for the covariances between the elements of $\hat{\boldsymbol{x}}$ is based on confidence ellipsoids. A confidence ellipsoid or error ellipsoid is described by the equation

$$\begin{aligned}
(\boldsymbol{x}-\hat{\boldsymbol{x}})^T\boldsymbol{Q}_{\hat{x}}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{x}}) &= q & (211)\\
\boldsymbol{y}^T(\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T)^{-1}\boldsymbol{y} &= q & (212)\\
\boldsymbol{y}^T\boldsymbol{V}\boldsymbol{\Lambda}^{-1}\boldsymbol{V}^T\boldsymbol{y} &= q & (213)\\
(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{V}^T\boldsymbol{y})^T(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{V}^T\boldsymbol{y}) &= q & (214)\\
(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{z})^T(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{z}) &= q & (215)\\
(z_1/\sqrt{\lambda_1})^2 + \cdots + (z_p/\sqrt{\lambda_p})^2 &= q & (216)\\
(z_1/\sqrt{q\,\lambda_1})^2 + \cdots + (z_p/\sqrt{q\,\lambda_p})^2 &= 1 & (217)
\end{aligned}$$

($q \geq 0$) where $\boldsymbol{V}$ is a matrix with the eigenvectors of $\boldsymbol{Q}_{\hat{x}} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^T$ in the columns (hence $\boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$) and $\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues of $\boldsymbol{Q}_{\hat{x}}$; $\boldsymbol{y} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ and $\boldsymbol{z} = \boldsymbol{V}^T\boldsymbol{y}$, $\boldsymbol{y} = \boldsymbol{V}\boldsymbol{z}$. This shows that the ellipsoid has semi axes in the directions of the eigenvectors and that their lengths are proportional to the square roots of the eigenvalues. The constant q depends on the confidence level and the distribution of the left hand side, see below. Since $\boldsymbol{Q}_{\hat{x}} = \sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}$ with known $\boldsymbol{A}$ and $\boldsymbol{P}$ we have two situations 1) σ_0^2 known and 2) σ_0^2 unknown.

σ_0^2 **known** In practice σ_0^2 is unknown so this case does not occur in the real world. If, however, σ_0^2 were known $(\boldsymbol{x}-\hat{\boldsymbol{x}})^T\boldsymbol{Q}_{\hat{x}}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{x}})$ would follow a χ^2 distribution with p degrees of freedom, $(\boldsymbol{x}-\hat{\boldsymbol{x}})^T\boldsymbol{Q}_{\hat{x}}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{x}}) \in \chi^2(p)$, and the semi axes of a, say, 95% confidence ellipsoid would be $\sqrt{q\,\lambda_i}$ where q is the 95% fractile of the $\chi^2(p)$ distribution and λ_i are the eigenvalues of $\boldsymbol{Q}_{\hat{x}}$.

σ_0^2 **unknown** In this case we estimate σ_0^2 as $\hat{\sigma}_0^2 = \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}/(n-p)$ which means that $(n-p)\hat{\sigma}_0^2 \in \chi^2(n-p)$. Also, $(\boldsymbol{x} - \hat{\boldsymbol{x}})^T(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})(\boldsymbol{x}-\hat{\boldsymbol{x}}) \in \chi^2(p)$. This means that

$$\frac{(\boldsymbol{x}-\hat{\boldsymbol{x}})^T(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})(\boldsymbol{x}-\hat{\boldsymbol{x}})/p}{\hat{\sigma}_0^2} \in F(p, n-p) \tag{218}$$

(since the independent numerator and denominator above follow $\chi^2(p)/p$ and $\chi^2(n-p)/(n-p)$ distributions, respectively. As n goes to infinity the above quantity multiplied by p approaches a $\chi^2(p)$ distribution so the above case with σ_0^2 known serves as a limiting case.) The semi axes of a, say, 95% confidence ellipsoid are $\sqrt{q\,p\,\lambda_i}$ where q is the 95% fractile of the $F(p, n-p)$ distribution, p is the number of parameters and λ_i are the eigenvalues of $\boldsymbol{Q}_{\hat{x}}$. If a subset of $m < p$ parameters are studied the semi axes of a, say, 95% confidence ellipsoid of the appropriate submatrix of $\boldsymbol{Q}_{\hat{x}}$ are $\sqrt{q\,m\,\lambda_i}$ where q is the 95% fractile of the $F(m, n-p)$ distribution, m is the number of parameters and λ_i are the eigenvalues of that submatrix, see also Examples 10 (page 32) and 11 (page 40) with Matlab code.

2.1.5 Dispersion of a Function of Estimated Parameters

To estimate the dispersion of some function f of the estimated parameters/elements (e.g. a distance determined by estimated coordinates) we perform a first order Taylor expansion around $\hat{\boldsymbol{x}}$

$$f(\boldsymbol{x}) \;\simeq\; f(\hat{\boldsymbol{x}}) + [\nabla f(\hat{\boldsymbol{x}})]^T(\boldsymbol{x} - \hat{\boldsymbol{x}}). \tag{219}$$

With $\boldsymbol{g} = \nabla f(\hat{\boldsymbol{x}})$ we get (again we use "=" instead of the correct "$\simeq$")

$$\mathrm{D}\{f\} = \sigma_0^2\, \boldsymbol{g}^T(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}\boldsymbol{g}, \tag{220}$$

see also Example 10 (page 34, example starts on page 32) with Matlab code.

2.1.6 The Derivative Matrix

The elements of the derivative matrix $\boldsymbol{A}$, $A_{ij} = \partial F_i/\partial x_j$, can be evaluated analytically or numerically.

Analytical partial derivatives for height or levelling observations are (z_A is the height in point A, z_B is the height in point B)

$$F = z_B - z_A \tag{221}$$

$$\frac{\partial F}{\partial z_A} = -1 \tag{222}$$

$$\frac{\partial F}{\partial z_B} = 1. \tag{223}$$

Equation 221 is obviously linear. If we do levelling only and don't combine with distance or directional observations we can do linear adjustment and we don't need the iterative procedure and the initial values for the elements. There are very few other geodesy, land surveying and GNSS related problems which can be solved by linear adjustment.

Analytical partial derivatives for 3-D distance observations are (remember that $d(\sqrt{u})/du = 1/(2\sqrt{u})$ and use the chain rule for differentiation)

$$F = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2} \tag{224}$$

$$= d_{AB} \tag{225}$$

$$\frac{\partial F}{\partial x_A} = \frac{1}{2d_{AB}} 2(x_B - x_A)(-1) \tag{226}$$

$$= -\frac{x_B - x_A}{d_{AB}} \tag{227}$$

$$\frac{\partial F}{\partial x_B} = -\frac{\partial F}{\partial x_A} \tag{228}$$

and similarly for y_A, y_B, z_A and z_B.

Analytical partial derivatives for 2-D distance observations are

$$
\begin{aligned}
F &= \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} && (229)\\
&= a_{AB} && (230)\\
\frac{\partial F}{\partial x_A} &= \frac{1}{2a_{AB}} 2(x_B - x_A)(-1) && (231)\\
&= -\frac{x_B - x_A}{a_{AB}} && (232)\\
\frac{\partial F}{\partial x_B} &= -\frac{\partial F}{\partial x_A} && (233)
\end{aligned}
$$

and similarly for y_A and y_B.

Analytical partial derivatives for horizontal direction observations are (remember that $d(\arctan u)/du = 1/(1+u^2)$ and again use the chain rule; $\arctan$ gives radians, r_A is in gon, $\omega = 200/\pi$ gon; r_A is related to the arbitrary zero for the horizontal direction measurement termed the orientation unknown[10])

$$
\begin{aligned}
F &= \omega \arctan \frac{y_B - y_A}{x_B - x_A} - r_A && (234)\\
\frac{\partial F}{\partial x_A} &= \omega \frac{1}{1 + (\frac{y_B - y_A}{x_B - x_A})^2} (-\frac{y_B - y_A}{(x_B - x_A)^2})(-1) && (235)\\
&= \omega \frac{y_B - y_A}{a_{AB}^2} && (236)\\
\frac{\partial F}{\partial x_B} &= -\frac{\partial F}{\partial x_A} && (237)\\
\frac{\partial F}{\partial y_A} &= \omega \frac{1}{1 + (\frac{y_B - y_A}{x_B - x_A})^2} \frac{1}{x_B - x_A}(-1) && (238)\\
&= -\omega \frac{x_B - x_A}{a_{AB}^2} && (239)\\
\frac{\partial F}{\partial y_B} &= -\frac{\partial F}{\partial y_A} && (240)\\
\frac{\partial F}{\partial r_A} &= -1. && (241)
\end{aligned}
$$

Numerical partial derivatives can be calculated as

$$
\begin{aligned}
\frac{\partial F(x_1, x_2, \ldots, x_p)}{\partial x_1} &\simeq \frac{F(x_1 + \delta, x_2, \ldots, x_p) - F(x_1, x_2, \ldots, x_p)}{\delta} && (242)\\
\frac{\partial F(x_1, x_2, \ldots, x_p)}{\partial x_2} &\simeq \frac{F(x_1, x_2 + \delta, \ldots, x_p) - F(x_1, x_2, \ldots, x_p)}{\delta} && (243)\\
&\vdots
\end{aligned}
$$

or we could use a symmetrized form

$$
\frac{\partial F(x_1, x_2, \ldots, x_p)}{\partial x_1} \simeq \frac{F(x_1 + \delta, x_2, \ldots, x_p) - F(x_1 - \delta, x_2, \ldots, x_p)}{2\delta} \qquad (244)
$$

[10] in Danish “kredsdrejningselement”

$$\frac{\partial F(x_1,x_2,\ldots,x_p)}{\partial x_2} \simeq \frac{F(x_1,x_2+\delta,\ldots,x_p)-F(x_1,x_2-\delta,\ldots,x_p)}{2\delta} \tag{245}$$

$$\vdots$$

both with δ appropriately small. Generally, one should be careful with numerical derivatives. There are two sources of error in the above equations, roundoff error that has to do with exact representation in the computer, and truncation error having to do with the magnitude of δ. In relation to Global Navigation Satellite System (GNSS) distance observations we are dealing with Fs with values larger than 20,000,000 meters (this is the approximate nadir distance from the GNSS space vehicles to the surface of the earth). In this connection a δ of 1 meter is small compared to F, it has an exact representation in the computer, and we don't have to do the division by δ (since it equals one). Note that when we use numerical partial derivatives we need $p+1$ function evaluations ($2p$ for the symmetrized form) for each iteration rather than one.

Example 10 (from Mærsk-Møller and Frederiksen, 1984, p. 86) This is a traditional land surveying example. From point 103 with unknown (2-D) coordinates we measure horizontal directions and distances to four points 016, 020, 015 and 013 (no distance is measured to point 020), see Figure 5. We wish to determine the coordinates of point 103 and the orientation unknown by means of nonlinear weighted least squares adjustment. The number of parameters is $p = 3$.

Points 016, 020, 015 and 013 are considered as error free fix points. Their coordinates are

Point	x [m]	y [m]
016	3725.10	3980.17
020	3465.74	4268.33
015	3155.96	4050.70
013	3130.55	3452.06

.

We measure four horizontal directions and three distances so we have seven observations, $n = 7$. Therefore we have $f = 7-3 = 4$ degrees of freedom. We determine the (2-D) coordinates $[x\ y]^T$ of point 103 and the the orientation unknown, r so $[x_1\ x_2\ x_3]^T = [x\ y\ r]^T$. The observation equations are (assuming that arctan gives radians and we want gon, $\omega = 200/\pi$ gon)

$$\ell_1 = \omega \arctan\frac{3980.17-y}{3725.10-x} - r + v_1 \tag{246}$$

$$\ell_2 = \omega \arctan\frac{4268.33-y}{3465.74-x} - r + v_2 \tag{247}$$

$$\ell_3 = \omega \arctan\frac{4050.70-y}{3155.96-x} - r + v_3 \tag{248}$$

$$\ell_4 = \omega \arctan\frac{3452.06-y}{3130.55-x} - r + v_4 \tag{249}$$

$$\ell_5 = \sqrt{(3725.10-x)^2+(3980.17-y)^2} + v_5 \tag{250}$$

$$\ell_6 = \sqrt{(3155.96-x)^2+(4050.70-y)^2} + v_6 \tag{251}$$

$$\ell_7 = \sqrt{(3130.55-x)^2+(3452.06-y)^2} + v_7. \tag{252}$$

We obtain the following observations (ℓ_i)

From	To	Horizontal	Horizontal
point	point	direction [gon]	distance [m]
103	016	0.000	706.260
103	020	30.013	
103	015	56.555	614.208
103	013	142.445	132.745

where the directional observations are means of two measurements. As the initial value $[x^*\ y^*]^T$ for the coordinates $[x\ y]^T$ of point 103 we choose the mean values for the coordinates of the four fix points. As the initial value r^* for the direction unknown r

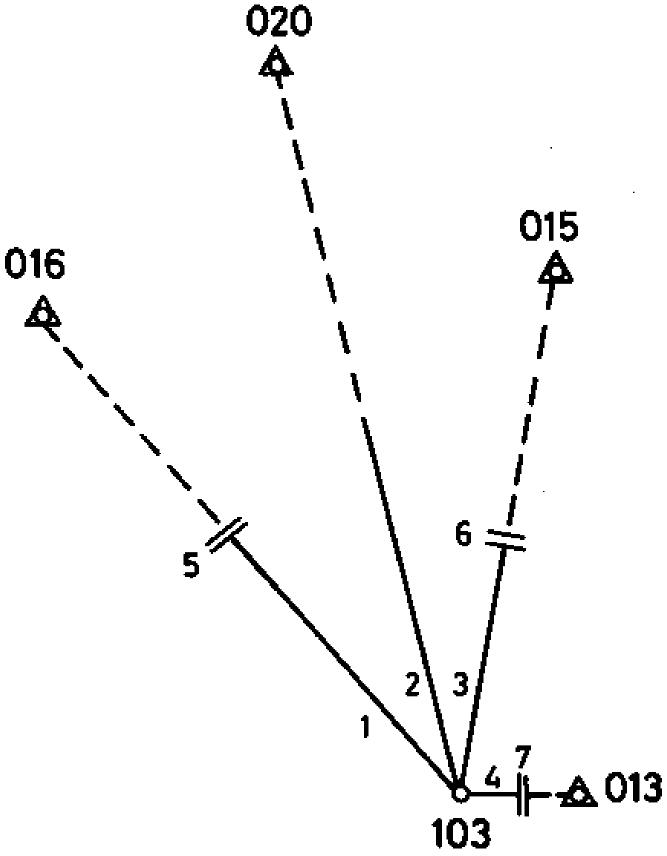

Figure 5: From point 103 with unknown coordinates we measure horizontal directions and distances (no distance is measured to point 020) to four points 016, 020, 015 and 013 (from Mærsk-Møller and Frederiksen, 1984; lefthand coordinate system).

we choose zero. First order Taylor expansions of the observation equations near the initial values give (assuming that $\arctan$ gives radians and we want gon; units for the first four equations are gon, for the last three units are m)

$$
\begin{aligned}
\ell_1 &= \omega \arctan \frac{3980.17 - y^*}{3725.10 - x^*} - r^* + \omega \frac{3980.17 - y^*}{a_1^2} \Delta_x - \omega \frac{3725.10 - x^*}{a_1^2} \Delta_y - \Delta_r + v_1 && (253)\\
\ell_2 &= \omega \arctan \frac{4268.33 - y^*}{3465.74 - x^*} - r^* + \omega \frac{3980.17 - y^*}{a_2^2} \Delta_x - \omega \frac{3725.10 - x^*}{a_2^2} \Delta_y - \Delta_r + v_2 && (254)\\
\ell_3 &= \omega \arctan \frac{4050.70 - y^*}{3155.96 - x^*} - r^* + \omega \frac{3980.17 - y^*}{a_3^2} \Delta_x - \omega \frac{3725.10 - x^*}{a_3^2} \Delta_y - \Delta_r + v_3 && (255)\\
\ell_4 &= \omega \arctan \frac{3452.06 - y^*}{3130.55 - x^*} - r^* + \omega \frac{3980.17 - y^*}{a_4^2} \Delta_x - \omega \frac{3725.10 - x^*}{a_4^2} \Delta_y - \Delta_r + v_4 && (256)\\
\ell_5 &= a_1 - \frac{3725.10 - x^*}{a_1} \Delta_x - \frac{3980.17 - y^*}{a_1} \Delta_y + v_5 && (257)\\
\ell_6 &= a_3 - \frac{3155.96 - x^*}{a_3} \Delta_x - \frac{4050.70 - y^*}{a_3} \Delta_y + v_6 && (258)\\
\ell_7 &= a_4 - \frac{3130.55 - x^*}{a_4} \Delta_x - \frac{3452.06 - y^*}{a_4} \Delta_y + v_7 && (259)
\end{aligned}
$$

where (units are m)

$$
\begin{aligned}
a_1 &= \sqrt{(3725.10 - x^*)^2 + (3980.17 - y^*)^2} && (260)\\
a_2 &= \sqrt{(3565.74 - x^*)^2 + (4268.33 - y^*)^2} && (261)\\
a_3 &= \sqrt{(3155.96 - x^*)^2 + (4050.70 - y^*)^2} && (262)\\
a_4 &= \sqrt{(3130.55 - x^*)^2 + (3452.06 - y^*)^2}. && (263)
\end{aligned}
$$

In matrix form we get ($\hat{\boldsymbol{k}} = \boldsymbol{A}\hat{\boldsymbol{\Delta}}$; as above units for the first four equations are gon, for the last three units are m)

$$\begin{bmatrix} 0.000 - \omega \arctan \frac{3980.17-y^*}{3725.10-x^*} + r^* \\ 30.013 - \omega \arctan \frac{4268.33-y^*}{3465.74-x^*} + r^* \\ 56.555 - \omega \arctan \frac{4050.70-y^*}{3155.96-x^*} + r^* \\ 142.445 - \omega \arctan \frac{3452.06-y^*}{3130.55-x^*} + r^* \\ 706.260 - a_1 \\ 614.208 - a_3 \\ 132.745 - a_4 \end{bmatrix} = \begin{bmatrix} \omega \frac{3980.17-y^*}{a_1^2} & -\omega \frac{3725.10-x^*}{a_1^2} & -1 \\ \omega \frac{3980.17-y^*}{a_2^2} & -\omega \frac{3725.10-x^*}{a_2^2} & -1 \\ \omega \frac{3980.17-y^*}{a_3^2} & -\omega \frac{3725.10-x^*}{a_3^2} & -1 \\ \omega \frac{3980.17-y^*}{a_4^2} & -\omega \frac{3725.10-x^*}{a_4^2} & -1 \\ -\frac{3725.10-x^*}{a_1} & -\frac{3980.17-y^*}{a_1} & 0 \\ -\frac{3155.96-x^*}{a_3} & -\frac{4050.70-y^*}{a_3} & 0 \\ -\frac{3130.55-x^*}{a_4} & -\frac{3452.06-y^*}{a_4} & 0 \end{bmatrix} \begin{bmatrix} \hat{\Delta}_x \\ \hat{\Delta}_y \\ \hat{\Delta}_r \end{bmatrix}. \tag{264}$$

The starting weight matrix is (for directions: $n = 2$, $s_c = 0.002$m, and $s_t = 0.0015$gon; for distances: $n = 1$, $s_G = 0.005$m, and $s_a = 0.005\text{m}/1000\text{m} = 0.000005$), see Section 1.2.2 (units for the first four weights are gon^{-2}, for the last three units are m^{-2})

$$\boldsymbol{P} = \begin{bmatrix} 0.7992 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.7925 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.7127 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.8472 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.03545 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.03780 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.03094 \end{bmatrix} \tag{265}$$

and after eleven iterations with the Matlab code below we end with (again, units for the first four weights are gon^{-2}, for the last three units are m^{-2})

$$\boldsymbol{P} = \begin{bmatrix} 0.8639 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.8714 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8562 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.4890 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.02669 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.02904 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.03931 \end{bmatrix}. \tag{266}$$

After the eleven iterations we get $[x\ y\ r]^T = [3,263.155\text{m}\ 3,445.925\text{m}\ 54.612\text{gon}]^T$ with standard deviations [4.14mm 2.49mm 0.641mgon]^T. The diagonal elements of the hat matrix $\boldsymbol{H}$ are [0.3629 0.3181 0.3014 0.7511 0.3322 0.2010 0.7332] and $p/n = 3/7 = 0.4286$. The diagonal elements of $\boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T$ are [0.4200mm^2 0.3650mm^2 0.3521mm^2 1.5360mm^2 12.4495mgon^2 6.9222mgon^2 18.6528mgon^2]. In this case where the observations have different units and therefore are on completely different scales, we need to look at the first four diagonal elements (representing the distance measurements) and the last three (representing the angle measurements) separately. The average of the first four is 0.6683mm^2, the average of the last three is 12.6748mgon^2. No observations have high leverages. Checking the diagonal of $\boldsymbol{A}\boldsymbol{N}^{-1}\boldsymbol{A}^T$ of course corresponds to checking the variances (or standard deviations) of the predicted observations $\hat{\boldsymbol{\ell}}$. The estimated residuals are $\hat{\boldsymbol{v}} = [-0.2352\text{mm}\ 0.9301\text{mm}\ -0.9171\text{mm}\ 0.3638\text{mm}\ -5.2262\text{mgon}\ 6.2309\text{mgon}\ -2.3408\text{mgon}]^T$. The resulting RMSE is $s_0 = 0.9563$. The probability of finding a larger value for RSS $= \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}$ is 0.4542 so s_0 is suitably small.

As an example on application of Equation 220 we calculate the distance between fix point 020 and point 103 and the standard deviation of the distance. From the Matlab code below we get the distance 846.989 m with a standard deviation of 2.66 mm.

The plots made in the code below allow us to study the iteration pattern of the Gauss-Newton method applied. The last plot produced, see Figure 6, shows the four fix points as triangles, the initial coordinates for point 103 as a plus, and the iterated solutions as circles marked with iteration number. The final solution is marked by both a plus and a circle. We see that since there are eleven iterations the last 3-4 iterations overlap in the plot.

A 95% confidence ellipsoid for $[x\ y\ r]^T$ with semi axes 18.47, 11.05 and 2.41 ($\sqrt{p\,6.591\,\lambda_i}$ where $p = 3$ is the number of parameters, 6.591 is the 95% fractile in the $F(3,4)$ distribution, and λ_i are the eigenvalues of $\boldsymbol{Q}_{\hat{x}} = \sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}$) is shown in Figure 7. Since the ellipsoid in the Matlab code in the notation of Section 2.1.4 in page 29 is generated in the $\boldsymbol{z}$-space we rotate by $\boldsymbol{V}$ to get to $\boldsymbol{y}$-space. [end of example]

Matlab code for Example 10

```
% (C) Copyright 2003-2004
% Allan Aasbjerg Nielsen
```

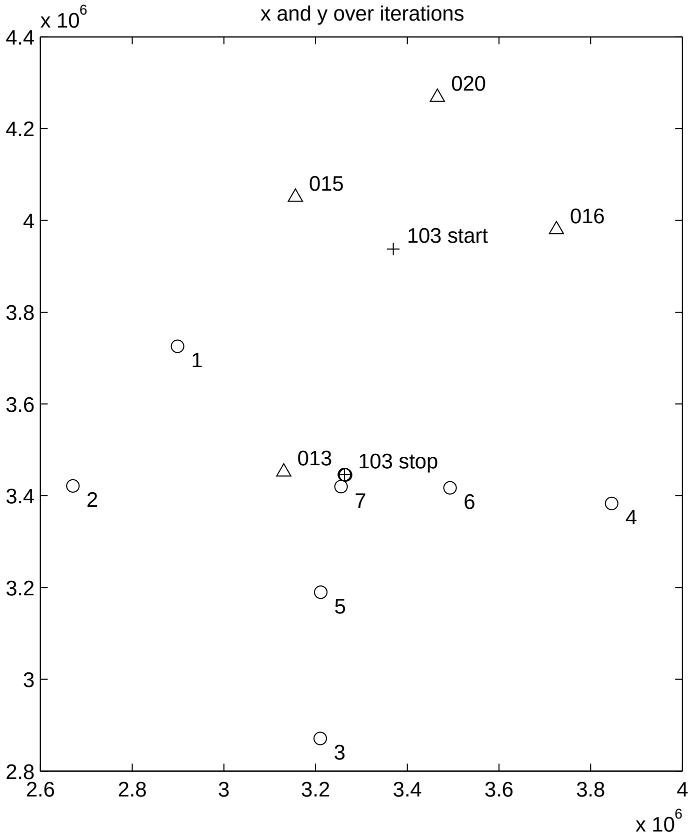

Figure 6: Development of x and y coordinates of point 103 over iterations with first seven iterations annotated; righthand coordinate system.

```
% aa@imm.dtu.dk, www.imm.dtu.dk/~aa

% analytical or numerical partial derivatives?
%partial = 'analytical';
partial = 'n';

cst = 200/pi; % radian to gon
eps = 0.001; % for numerical differentiation

% positions of points 016, 020, 015 and 013 in network, [m]
xx = [3725.10 3465.74 3155.96 3130.55]';
yy = [3980.17 4268.33 4050.70 3452.06]';

% observations: 1-4 are directions [gon], 5-7 are distances [m]
l = [0 30.013 56.555 142.445 706.260 614.208 132.745]'; % l is \ell (not one)
n = size(l,1);

% initial values for elements: x- and y-coordinates for point 103 [m], and
% the direction unknown [gon]
x = [3263.150 3445.920 54.6122]';
% play with initial values to check robustness of method
x = [0 0 -200]';
x = [0 0 -100]';
x = [0 0  100]';
x = [0 0  200]';
x = [0 0  40000]';
x = [0 0    0]';
x = [100000 100000 0]';
x = [mean(xx) mean(yy) 0]';
%x = [mean(xx) 3452.06 0]'; % approx. same y as 013
p = size(x,1);

% desired units: mm and mgon
xx = 1000*xx;
yy = 1000*yy;
```

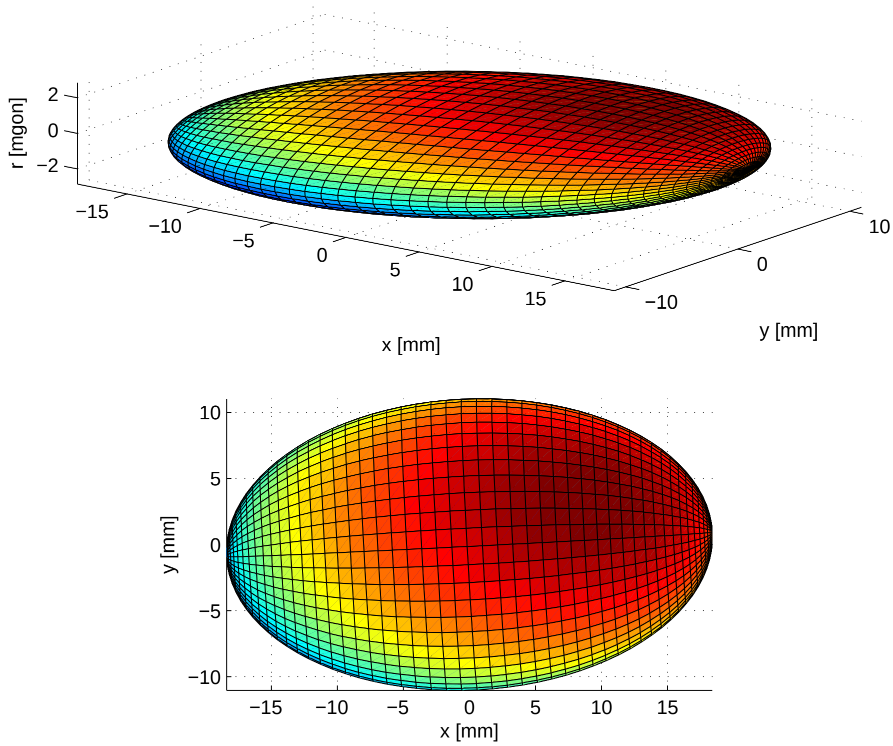

Figure 7: 95% ellipsoid for $[x\ y\ r]^T$ with projection on xy-plane.

```
l = 1000*l;
x = 1000*x;
cst = 1000*cst;

%number of degrees of freedom
f = n-p;

x0 = x;

sc = 0.002*1000;%[mm]
st = 0.0015*1000;%[mgon]
sG = 0.005*1000;%[mm]
sa = 0.000005;%[m/m], no unit
%a [mm]

idx = [];
e2 = [];
dc = [];
X = [];

for iter = 1:50 % iter ----------------------------------------------------------

% output from atan2 is in radian, convert to gon
F1 = cst.*atan2(yy-x(2),xx-x(1))-x(3);
a = (x(1)-xx).^2+(x(2)-yy).^2;
F2 = sqrt(a);
F = [F1; F2([1 3:end])]; % skip distance from 103 to 020

% weight matrix
%a [mm]
P = diag([2./(2*(cst*sc).^2./a+st^2); 1./(sG^2+a([1 3:end])*sa.^2)]);
diag(P)’

k = l-F; % l is \ell (not one)

A1 = [];
A2 = [];
```

```
if strcmp(partial,'analytical')
    % A is matrix of analytical partial derivatives
    error('not implemented yet');
else
    % A is matrix of numerical partial derivatives
    %directions
    dF = (cst.*atan2(yy- x(2)     ,xx-(x(1)+eps))- x(3)   -F1)/eps;
    A1 = [A1 dF];
    dF = (cst.*atan2(yy-(x(2)+eps),xx- x(1)    ) - x(3)   -F1)/eps;
    A1 = [A1 dF];
    dF = (cst.*atan2(yy- x(2)      ,xx- x(1)    ) -(x(3)+eps)-F1)/eps;
    A1 = [A1 dF];
    %distances
    dF = (sqrt((x(1)+eps-xx).^2+(x(2)    -yy).^2)-F2)/eps;
    A2 = [A2 dF];
    dF = (sqrt((x(1)    -xx).^2+(x(2)+eps-yy).^2)-F2)/eps;
    A2 = [A2 dF];
    dF = (sqrt((x(1)    -xx).^2+(x(2)    -yy).^2)-F2)/eps;
    A2 = [A2 dF];
    A2 = A2([1 3:4],:);% skip derivatives of distance from 103 to 020

    A = [A1; A2];
end

N = A'*P;
c = N*k;
N = N*A;
%WLS
deltahat = N\c;
khat = A*deltahat;
vhat = k-khat;
e2 = [e2 vhat'*P*vhat];
dc = [dc deltahat'*c];
%update for iterations
x = x+deltahat;
X = [X x];

idx = [idx iter];

% stop iterations
itertst = (k'*P*k)/e2(end);
if itertst < 1.000001
    break;
end

end % iter ------------------------------------------------------------------

%x-x0
% number of iterations
iter

%MSE
s02 = e2(end)/f;
%RMSE
s0 = sqrt(s02)

%Variance/covariance matrix of the observations, l
Dl = s02.*inv(P);
%Standard deviations
stdl = sqrt(diag(Dl))

%Variance/covariance matrix of the adjusted elements, xhat
Ninv = inv(N);
Dxhat = s02.*Ninv;
%Standard deviations
stdxhat = sqrt(diag(Dxhat))

%Variance/covariance matrix of the adjusted observations, lhat
Dlhat = s02.*A*Ninv*A';
%Standard deviations
stdlhat = sqrt(diag(Dlhat))

%Variance/covariance matrix of the adjusted residuals, vhat
Dvhat = Dl-Dlhat;
```

```
%Standard deviations
stdvhat = sqrt(diag(Dvhat))

%Correlations between adjusted elements, xhat
aux = diag(1./stdxhat);
corrxhat = aux*Dxhat*aux

% Standard deviation of estimated distance from 103 to 020
d020 = sqrt((xx(2)-x(1))^2+(yy(2)-x(2))^2);
%numerical partial derivatives of d020, i.e. gradient of d020
grad = [];
dF = (sqrt((xx(2)-(x(1)+eps))^2+(yy(2)- x(2)     )^2)-d020)/eps;
grad = [grad; dF];
dF = (sqrt((xx(2)- x(1)     )^2+(yy(2)-(x(2)+eps))^2)-d020)/eps;
grad = [grad; dF; 0];
stdd020 = s0*sqrt(grad’*Ninv*grad)

% plots to illustrate progress of iterations
figure
%plot(idx,e2);
semilogy(idx,e2);
title(’v^TPv’);
figure
%plot(idx,dc);
semilogy(idx,dc);
title(’c^TN^{-1}c’);
figure
%plot(idx,dc./e2);
semilogy(idx,dc./e2);
title(’(c^TN^{-1}c)/(v^TPv)’);
for i = 1:p
    figure
    plot(idx,X(i,:));
    %semilogy(idx,X(i,:));
    title(’X(i,:) vs. iteration index’);
end
figure
%loglog(x0(1),x0(2),’k+’)
plot(x0(1),x0(2),’k+’) % initial values for x and y
text(x0(1)+30000,x0(2)+30000,’103 start’)
hold
% positions of points 016, 020, 015 and 013 in network
%loglog(xx,yy,’xk’)
plot(xx,yy,’k^’)
txt = [’016’; ’020’; ’015’; ’013’];
for i = 1:4
    text(xx(i)+30000,yy(i)+30000,txt(i,:));
end
for i = 1:iter
%    loglog(X(1,i),X(2,i),’ko’);
    plot(X(1,i),X(2,i),’ko’);
end
for i = 1:7
    text(X(1,i)+30000,X(2,i)-30000,num2str(i));
end
plot(X(1,end),X(2,end),’k+’);
%loglog(X(1,end),X(2,end),’k+’);
text(X(1,end)+30000,X(2,end)+30000,’103 stop’)
title(’x and y over iterations’);
%title(’X(1,:) vs. X(2,:) over iterations’);
axis equal
axis([2.6e6 4e6 2.8e6 4.4e6])

% t-values and probabilities of finding larger |t|
% pt should be smaller than, say, (5% or) 1%
t = x./stdxhat;
pt = betainc(f./(f+t.^2),0.5*f,0.5);

% probabilitiy of finding larger v’Pv
% should be greater than, say, 5% (or 1%)
pchi2 = 1-gammainc(0.5*e2(end),0.5*f);

% semi-axes in confidence ellipsoid
% 95% fractile for 3 dfs is  7.815 = 2.796^2
% 99% fractile for 3 dfs is 11.342 = 3.368^2
%[vDxhat dDxhat] = eigsort(Dxhat(1:2,1:2));
[vDxhat dDxhat] = eigsort(Dxhat);
%semiaxes = sqrt(diag(dDxhat));
% 95% fractile for 2 dfs is  5.991 = 2.448^2
% 99% fractile for 2 dfs is  9.210 = 3.035^2
%   df    F(3,df).95  F(3,df).99
%    1      215.71      5403.1
%    2       19.164       99.166
%    3        9.277       29.456
%    4        6.591       16.694
%    5        5.409       12.060
%   10        3.708        6.552
%  100        2.696        3.984
%  inf        2.605        3.781
% chi^2 approximation, 95% fractile
semiaxes = sqrt(7.815*diag(dDxhat))
figure
ellipsoidrot(0,0,0,semiaxes(1),semiaxes(2),semiaxes(3),vDxhat);
axis equal
xlabel('x [mm]'); ylabel('y [mm]'); zlabel('r [mgon]');
title('95% confidence ellipsoid, \chi^2 approx.')
% F approximation, 95% fractile. NB the fractile depends on df
semiaxes = sqrt(3*6.591*diag(dDxhat))
figure
ellipsoidrot(0,0,0,semiaxes(1),semiaxes(2),semiaxes(3),vDxhat);
axis equal
xlabel('x [mm]'); ylabel('y [mm]'); zlabel('r [mgon]');
title('95% confidence ellipsoid, F approx.')
view(37.5,15)
print -depsc2 confxyr.eps
%clear
%close all

function [v,d] = eigsort(a)
[v1,d1] = eig(a);
d2 = diag(d1);
[dum,idx] = sort(d2);
v = v1(:,idx);
d = diag(d2(idx));

function [xx,yy,zz] = ellipsoidrot(xc,yc,zc,xr,yr,zr,Q,n)
%ELLIPSOID Generate ellipsoid.
%
% [X,Y,Z] = ELLIPSOID(XC,YC,ZC,XR,YR,ZR,Q,N) generates three
% (N+1)-by-(N+1) matrices so that SURF(X,Y,Z) produces a rotated
% ellipsoid with center (XC,YC,ZC) and radii XR, YR, ZR.
%
% [X,Y,Z] = ELLIPSOID(XC,YC,ZC,XR,YR,ZR,Q) uses N = 20.
%
% ELLIPSOID(...) and ELLIPSOID(...,N) with no output arguments
% graph the ellipsoid as a SURFACE and do not return anything.
%
% The ellipsoidal data is generated using the equation after rotation with
% orthogonal matrix Q:
%
%  (X-XC)^2     (Y-YC)^2     (Z-ZC)^2
%  --------  +  --------  +  --------  =  1
%    XR^2         YR^2         ZR^2
%
% See also SPHERE, CYLINDER.

% Modified by Allan Aasbjerg Nielsen (2004) after
% Laurens Schalekamp and Damian T. Packer
% Copyright 1984-2002 The MathWorks, Inc.
% $Revision: 1.7 $  $Date: 2002/06/14 20:33:49 $

error(nargchk(7,8,nargin));

if nargin == 7
n = 20;
end

[x,y,z] = sphere(n);
```

```
x = xr*x;
y = yr*y;
z = zr*z;
xvec = Q*[reshape(x,1,(n+1)^2); reshape(y,1,(n+1)^2); reshape(z,1,(n+1)^2)];
x = reshape(xvec(1,:),n+1,n+1)+xc;
y = reshape(xvec(2,:),n+1,n+1)+yc;
z = reshape(xvec(3,:),n+1,n+1)+zc;
if(nargout == 0)
    surf(x,y,z)
%   surfl(x,y,z)
%   surfc(x,y,z)
    axis equal
    %shading interp
    %colormap gray
else
xx = x;
yy = y;
zz = z;
end
```

Example 11 In this example we have data on the positions of Navstar Global Positioning System (GPS) space vehicles (SV) 1, 4, 7, 13, 20, 24 and 25 and pseudoranges from our position to the SVs. We want to determine the (3-D) coordinates $[X\ Y\ Z]^T$ of our position and the clock error in our GPS receiver, cdT, $[x_1\ x_2\ x_3\ x_4]^T = [X\ Y\ Z\ cdT]^T$, so the number of parameters is $p = 4$. The positions of and pseudoranges (ℓ) to the SVs given in a data file from the GPS receiver are

SV	X [m]	Y [m]	Z [m]	ℓ [m]
1	16,577,402.072	5,640,460.750	20,151,933.185	20,432,524.0
4	11,793,840.229	–10,611,621.371	21,372,809.480	21,434,024.4
7	20,141,014.004	–17,040,472.264	2,512,131.115	24,556,171.0
13	22,622,494.101	–4,288,365.463	13,137,555.567	21,315,100.2
20	12,867,750.433	15,820,032.908	16,952,442.746	21,255,217.0
24	–3,189,257.131	–17,447,568.373	20,051,400.790	24,441,547.2
25	–7,437,756.358	13,957,664.984	21,692,377.935	23,768,678.3

.

The true position is (that of the no longer existing GPS station at Landmålervej in Hjortekær) $[X\ Y\ Z]^T = [3,507,884.948\ 780,492.718\ 5,251,780.403]^T$ m. We have seven observations, $n = 7$. Therefore we have $f = n - p = 7 - 4 = 3$ degrees of freedom. The observation equations are (in m)

$$\ell_1 = \sqrt{(16577402.072 - X)^2 + (5640460.750 - Y)^2 + (20151933.185 - Z)^2} + cdT + v_1 \tag{267}$$

$$\ell_2 = \sqrt{(11793840.229 - X)^2 + (-10611621.371 - Y)^2 + (21372809.480 - Z)^2} + cdT + v_2 \tag{268}$$

$$\ell_3 = \sqrt{(20141014.004 - X)^2 + (-17040472.264 - Y)^2 + (2512131.115 - Z)^2} + cdT + v_3 \tag{269}$$

$$\ell_4 = \sqrt{(22622494.101 - X)^2 + (-4288365.463 - Y)^2 + (13137555.567 - Z)^2} + cdT + v_4 \tag{270}$$

$$\ell_5 = \sqrt{(12867750.433 - X)^2 + (15820032.908 - Y)^2 + (16952442.746 - Z)^2} + cdT + v_5 \tag{271}$$

$$\ell_6 = \sqrt{(-3189257.131 - X)^2 + (-17447568.373 - Y)^2 + (20051400.790 - Z)^2} + cdT + v_6 \tag{272}$$

$$\ell_7 = \sqrt{(-7437756.358 - X)^2 + (13957664.984 - Y)^2 + (21692377.935 - Z)^2} + cdT + v_7 \tag{273}$$

As the initial values $[X^*\ Y^*\ Z^*\ cdT^*]^T$ we choose $[0\ 0\ 0\ 0]^T$, center of the Earth, no clock error. First order Taylor expansions of the observation equations near the initial values give (in m)

$$\ell_1 = d_1 + cdT^* \tag{274}$$

$$-\frac{16577402.072 - X^*}{d_1}\Delta_X - \frac{5640460.750 - Y^*}{d_1}\Delta_Y - \frac{20151933.185 - Z^*}{d_1}\Delta_Z + \Delta_{cdT} + v_1$$

$$\begin{aligned} \ell_2 = {} & d_2 + cdT^* \\ & -\frac{11793840.229 - X^*}{d_2}\Delta_X - \frac{-10611621.371 - Y^*}{d_2}\Delta_Y - \frac{21372809.480 - Z^*}{d_2}\Delta_Z + \Delta_{cdT} + v_2 \end{aligned} \tag{275}$$

$$\begin{aligned} \ell_3 = {} & d_3 + cdT^* \\ & -\frac{20141014.004 - X^*}{d_3}\Delta_X - \frac{-17040472.264 - Y^*}{d_3}\Delta_Y - \frac{2512131.115 - Z^*}{d_3}\Delta_Z + \Delta_{cdT} + v_3 \end{aligned} \tag{276}$$

$$\begin{aligned} \ell_4 = {} & d_4 + cdT^* \\ & -\frac{22622494.101 - X^*}{d_4}\Delta_X - \frac{-4288365.463 - Y^*}{d_4}\Delta_Y - \frac{13137555.567 - Z^*}{d_4}\Delta_Z + \Delta_{cdT} + v_4 \end{aligned} \tag{277}$$

$$\begin{aligned} \ell_5 = {} & d_5 + cdT^* \\ & -\frac{12867750.433 - X^*}{d_5}\Delta_X - \frac{15820032.908 - Y^*}{d_5}\Delta_Y - \frac{16952442.746 - Z^*}{d_5}\Delta_Z + \Delta_{cdT} + v_5 \end{aligned} \tag{278}$$

$$\begin{aligned} \ell_6 = {} & d_6 + cdT^* \\ & -\frac{-3189257.131 - X^*}{d_6}\Delta_X - \frac{-17447568.373 - Y^*}{d_6}\Delta_Y - \frac{20051400.790 - Z^*}{d_6}\Delta_Z + \Delta_{cdT} + v_6 \end{aligned} \tag{279}$$

$$\begin{aligned} \ell_7 = {} & d_7 + cdT^* \\ & -\frac{-7437756.358 - X^*}{d_7}\Delta_X - \frac{13957664.984 - Y^*}{d_7}\Delta_Y - \frac{21692377.935 - Z^*}{d_7}\Delta_Z + \Delta_{cdT} + v_7 \end{aligned} \tag{280}$$

where (in m)

$$d_1 = \sqrt{(16577402.072 - X^*)^2 + (5640460.750 - Y^*)^2 + (20151933.185 - Z^*)^2} \tag{281}$$

$$d_2 = \sqrt{(11793840.229 - X^*)^2 + (-10611621.371 - Y^*)^2 + (21372809.480 - Z^*)^2} \tag{282}$$

$$d_3 = \sqrt{(20141014.004 - X^*)^2 + (-17040472.264 - Y^*)^2 + (2512131.115 - Z^*)^2} \tag{283}$$

$$d_4 = \sqrt{(22622494.101 - X^*)^2 + (-4288365.463 - Y^*)^2 + (13137555.567 - Z^*)^2} \tag{284}$$

$$d_5 = \sqrt{(12867750.433 - X^*)^2 + (15820032.908 - Y^*)^2 + (16952442.746 - Z^*)^2} \tag{285}$$

$$d_6 = \sqrt{(-3189257.131 - X^*)^2 + (-17447568.373 - Y^*)^2 + (20051400.790 - Z^*)^2} \tag{286}$$

$$d_7 = \sqrt{(-7437756.358 - X^*)^2 + (13957664.984 - Y^*)^2 + (21692377.935 - Z^*)^2}. \tag{287}$$

In matrix form we get ($\hat{\boldsymbol{k}} = \boldsymbol{A}\hat{\boldsymbol{\Delta}}$; as above units are m)

$$\begin{bmatrix} 20432524.0 - d_1 - cdT^* \\ 21434024.4 - d_2 - cdT^* \\ 24556171.0 - d_3 - cdT^* \\ 21315100.2 - d_4 - cdT^* \\ 21255217.0 - d_5 - cdT^* \\ 24441547.2 - d_6 - cdT^* \\ 23768678.3 - d_7 - cdT^* \end{bmatrix} = \begin{bmatrix} -\frac{16577402.072 - X^*}{d_1} & -\frac{5640460.750 - Y^*}{d_1} & -\frac{20151933.185 - Z^*}{d_1} & 1 \\ -\frac{11793840.229 - X^*}{d_2} & -\frac{-10611621.371 - Y^*}{d_2} & -\frac{21372809.480 - Z^*}{d_2} & 1 \\ -\frac{20141014.004 - X^*}{d_3} & -\frac{-17040472.264 - Y^*}{d_3} & -\frac{2512131.115 - Z^*}{d_3} & 1 \\ -\frac{22622494.101 - X^*}{d_4} & -\frac{-4288365.463 - Y^*}{d_4} & -\frac{13137555.567 - Z^*}{d_4} & 1 \\ -\frac{12867750.433 - X^*}{d_5} & -\frac{15820032.908 - Y^*}{d_5} & -\frac{16952442.746 - Z^*}{d_5} & 1 \\ -\frac{-3189257.131 - X^*}{d_6} & -\frac{-17447568.373 - Y^*}{d_6} & -\frac{20051400.790 - Z^*}{d_6} & 1 \\ -\frac{-7437756.358 - X^*}{d_7} & -\frac{13957664.984 - Y^*}{d_7} & -\frac{21692377.935 - Z^*}{d_7} & 1 \end{bmatrix} \begin{bmatrix} \hat{\Delta}_X \\ \hat{\Delta}_Y \\ \hat{\Delta}_Z \\ \hat{\Delta}_{cdT} \end{bmatrix} \tag{288}$$

After five iterations with the Matlab code below (with all observations weighted equally, $p_i = 1/(10\text{m})^2$) we get $[X\ Y\ Z\ cdT]^T = [3,507,889.1\ \ 780,490.0\ \ 5,251,783.8\ \ 25,511.1]^T$ m with standard deviations $[6.42\ 5.31\ 11.69\ 7.86]^T$ m. $25,511.1$ m corresponds to a clock error of 0.085 ms. The difference between the true position and the solution found is $[-4.18\ \ 2.70\ \ -3.35]^T$ m, all well within one standard deviation. The corresponding distance is 6.00 m. Figure 8 shows the four parameters over the iterations including the starting guess. The diagonal elements of the hat matrix $\boldsymbol{H}$ are $[0.4144\ 0.5200\ 0.8572\ 0.3528\ 0.4900\ 0.6437\ 0.7218]$ and $p/n = 4/7 = 0.5714$ so no observations have high leverages. The estimated residuals are $\hat{\boldsymbol{v}} = [5.80\ -5.10\ 0.74\ -5.03\ 3.20\ 5.56\ -5.17]^T$ m. With prior variances of 10^2 m^2, the resulting RMSE is $s_0 = 0.7149$.

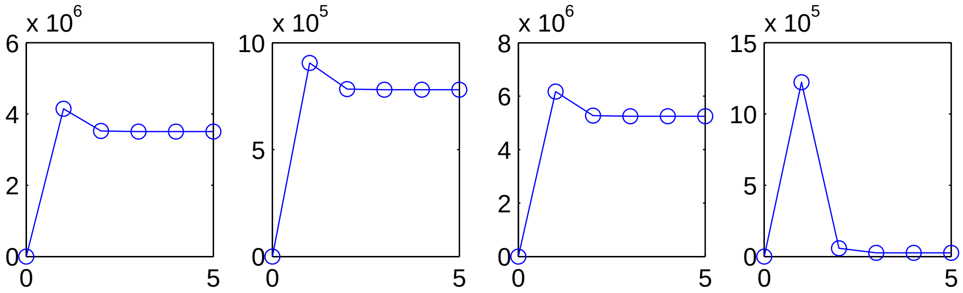

Figure 8: Parameters $[X\ Y\ Z\ cdT]^T$ over iterations including the starting guess.

The probability of finding a larger value for RSS $= \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}$ is 0.6747 so s_0 is suitably small. With prior variances of 5^2 m^2 instead of 10^2 m^2, s_0 is 1.4297 and the probability of finding a larger value for RSS $= \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}$ is 0.1054 so also in this situation s_0 is suitably small. With prior variances of 3^2 m^2 instead of 10^2 m^2, s_0 is 2.3828 and the probability of finding a larger value for RSS $= \hat{\boldsymbol{v}}^T\boldsymbol{P}\hat{\boldsymbol{v}}$ is 0.0007 so in this situation s_0 is too large.

95% confidence ellipsoids for $[X\ Y\ Z]^T$ in an earth-centered-earth-fixed (ECEF) coordinate system and in a local Easting-Northing-Up (ENU) coordinate system are shown in Figure 9. The semi axes in both the ECEF and the ENU systems are 64.92, 30.76 and 23.96 (this is $\sqrt{m\ 9.277\ \lambda_i}$ where $m = 3$ is the number of parameters, 9.277 is the 95% fractile in the $F(3,3)$ distribution, and λ_i are the eigenvalues of $\boldsymbol{Q}_{XYZ}$, the upper-left 3×3 submatrix of $\boldsymbol{Q}_{\hat{x}} = \sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}$); units are metres. The rotation to the local ENU system is performed by means of the orthogonal matrix ($\boldsymbol{F}^T\boldsymbol{F} = \boldsymbol{F}\boldsymbol{F}^T = \boldsymbol{I}$)

$$\boldsymbol{F}^T = \begin{bmatrix} -\sin\lambda & \cos\lambda & 0 \\ -\sin\phi\cos\lambda & -\sin\phi\sin\lambda & \cos\phi \\ \cos\phi\cos\lambda & \cos\phi\sin\lambda & \sin\phi \end{bmatrix} \tag{289}$$

where ϕ is the latitude and λ is the longitude. The variance-covariance matrix of the position estimates in the ENU coordinate system is $\boldsymbol{Q}_{ENU} = \boldsymbol{F}^T\boldsymbol{Q}_{XYZ}\boldsymbol{F}$. Since

$$\boldsymbol{Q}_{XYZ}\boldsymbol{a} = \lambda\boldsymbol{a} \tag{290}$$

$$\boldsymbol{F}^T\boldsymbol{Q}_{XYZ}\boldsymbol{F}\boldsymbol{F}^T\boldsymbol{a} = \lambda\boldsymbol{F}^T\boldsymbol{a} \tag{291}$$

$$\boldsymbol{Q}_{ENU}(\boldsymbol{F}^T\boldsymbol{a}) = \lambda(\boldsymbol{F}^T\boldsymbol{a}) \tag{292}$$

we see that $\boldsymbol{Q}_{XYZ}$ and $\boldsymbol{Q}_{ENU}$ have the same eigenvalues and their eigenvectors are related as indicated. Since the ellipsoid in the Matlab code in the notation of Section 2.1.4 in page 29 is generated in the $\boldsymbol{z}$-space we rotate by $\boldsymbol{V}$ to get to $\boldsymbol{y}$-space.

Dilution of Precision, DOP Satellite positioning works best when there is a good angular separation between the space vehicles. A measure of this separation is termed the dilution of precision, DOP. Low values of the DOP correspond to a good angular separation, high values to a bad angular separation, i.e., a high degree of clustering of the SVs. There are several versions of DOP. From Equation 206 the dispersion of the parameters is

$$\boldsymbol{Q}_{\hat{x}} = \mathrm{D}\{\hat{\boldsymbol{x}}_{WLS}\} = \sigma_0^2(\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}. \tag{293}$$

This matrix has contributions from our prior expectations to the precision of the measurements ($\boldsymbol{P}$), the actual precision of the measurements (σ_0^2) and the geometry of the problem ($\boldsymbol{A}$). Let's look at the geometry alone and define the symmetric matrix

$$\boldsymbol{Q}_{DOP} = \boldsymbol{Q}_{\hat{x}}/(\sigma_0^2\ \sigma_{prior}^2) = (\boldsymbol{A}^T\boldsymbol{P}\boldsymbol{A})^{-1}/\sigma_{prior}^2 = \begin{bmatrix} q_X^2 & q_{XY} & q_{XZ} & q_{XcdT} \\ q_{XY} & q_Y^2 & q_{YZ} & q_{YcdT} \\ q_{XZ} & q_{YZ} & q_Z^2 & q_{ZcdT} \\ q_{XcdT} & q_{YcdT} & q_{ZcdT} & q_{cdT}^2 \end{bmatrix} \tag{294}$$

where $\sigma_{prior}^2 = \sigma_{i,prior}^2$, i.e., all prior variances are equal, see Section 1.2.2. In WLS (with equal weights on all observations) this corresponds to $\boldsymbol{Q}_{DOP} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}$.

We are now ready to define the position DOP

$$PDOP = \sqrt{q_X^2 + q_Y^2 + q_Z^2}, \tag{295}$$

the time DOP

$$TDOP = \sqrt{q_{cdT}^2} = q_{cdT} \tag{296}$$

and the geometric DOP

$$GDOP = \sqrt{q_X^2 + q_Y^2 + q_Z^2 + q_{cdT}^2} \tag{297}$$

which is the square root of the trace of $\boldsymbol{Q}_{DOP}$. It is easily seen that $GDOP^2 = PDOP^2 + TDOP^2$.

In practice PDOP values less than 2 are considered excellent, between 2 and 4 good, up to 6 acceptable. PDOP values greater than around 6 are considered suspect.

DOP is a measure of size of the matrix $\boldsymbol{Q}_{DOP}$ (or sub-matrices thereof, for PDOP for example the upper left three by three matrix). As an alternative measure of this size we could use the determinant. Such a DOP measure would allow for off-diagonal elements of $\boldsymbol{Q}_{DOP}$, i.e., for covariances between the final estimates of the position. Determinant based DOP measures are not used in the GNSS literature.

After rotation from the ECEF to the ENU coordinate system which transforms the upper-left 3 × 3 submatrix $\boldsymbol{Q}_{XYZ}$ of $\boldsymbol{Q}_{\hat{x}}$ into $\boldsymbol{Q}_{ENU}$, we can define

$$\boldsymbol{Q}_{DOP,ENU} = \boldsymbol{Q}_{ENU}/(\sigma_0^2\,\sigma_{prior}^2) = \begin{bmatrix} q_E^2 & q_{EN} & q_{EU} \\ q_{EN} & q_N^2 & q_{NU} \\ q_{EU} & q_{NU} & q_U^2 \end{bmatrix}, \tag{298}$$

the horizontal DOP

$$HDOP = \sqrt{q_E^2 + q_N^2} \tag{299}$$

and the vertical DOP

$$VDOP = \sqrt{q_U^2} = q_U. \tag{300}$$

We see that $PDOP^2 = HDOP^2 + VDOP^2$ which is the trace of $\boldsymbol{Q}_{DOP,ENU}$. [end of example]

Matlab code for Example 11

Code for functions `eigsort` and `ellipsoidrot` are listed under Example 10.

```
% (C) Copyright 2004
% Allan Aasbjerg Nielsen
% aa@imm.dtu.dk, www.imm.dtu.dk/~aa

format short g

% use analytical partial derivatives
partial = 'analytical';
%partial = 'n';
% speed of light, [m/s]
%clight = 300000000;
clight = 299792458;
% length of C/A code, [m]
%L = 300000;

% true position (Landmaalervej, Hjortekaer)
xtrue = [3507884.948 780492.718 5251780.403 0]';
% positions of satellites 1, 4, 7, 13, 20, 24 and 25 in ECEF coordinate system, [m]
xxyyzz = [16577402.072   5640460.750 20151933.185;
          11793840.229 -10611621.371 21372809.480;
          20141014.004 -17040472.264  2512131.115;
          22622494.101  -4288365.463 13137555.567;
          12867750.433  15820032.908 16952442.746;
          -3189257.131 -17447568.373 20051400.790;
```

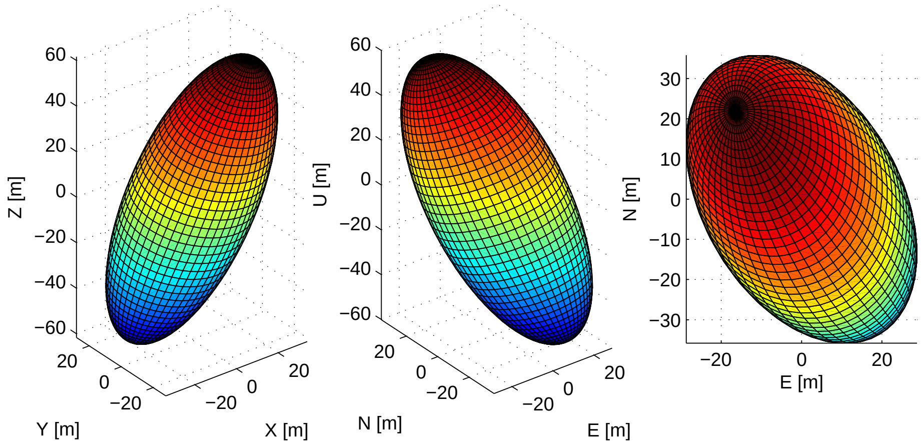

Figure 9: 95% ellipsoid for $[X\ Y\ Z]^T$ in ECEF (left) and ENU (middle) coordinate systems with projection on EN-plane (right).

```
          -7437756.358  13957664.984 21692377.935];
pseudorange = [20432524.0 21434024.4 24556171.0 21315100.2 21255217.0 ...
    24441547.2 23768678.3]’; % [m]
l = pseudorange; % l is \ell (not one)
xx = xxyyzz(:,1);
yy = xxyyzz(:,2);
zz = xxyyzz(:,3);
n = size(xx,1); % number of observations

% weight matrix
sprior2 = 10^2; %5^2; %prior variance [m^2]
P = eye(n)/sprior2; % weight = 1/"prior variance" [m^(-2)]

% preliminary position, [m]
x = [0 0 0 0]’;
p = size(x,1); % number of elements/parameters
f = n-p; % number of degrees of freedom
x0 = x;

for iter = 1:20 % iter -----------------------------------------------------------

range = sqrt((x(1)-xx).^2+(x(2)-yy).^2+(x(3)-zz).^2);
prange = range+x(4);
F = prange;

A = [];
if strcmp(partial,’analytical’)
    % A is matrix of analytical partial derivatives
    irange = 1./range;
    dF = irange.*(x(1)-xx);
    A = [A dF];
    dF = irange.*(x(2)-yy);
    A = [A dF];
    dF = irange.*(x(3)-zz);
    A = [A dF];
    dF = ones(n,1);
    A = [A dF];
else
    % A is matrix of numerical partial derivatives
    dF = sqrt((x(1)+1-xx).^2+(x(2)  -yy).^2+(x(3)  -zz).^2)+ x(4)   -prange;
    A = [A dF];
    dF = sqrt((x(1)  -xx).^2+(x(2)+1-yy).^2+(x(3)  -zz).^2)+ x(4)   -prange;
    A = [A dF];
    dF = sqrt((x(1)  -xx).^2+(x(2)  -yy).^2+(x(3)+1-zz).^2)+ x(4)   -prange;
    A = [A dF];
    dF = sqrt((x(1)  -xx).^2+(x(2)  -yy).^2+(x(3)  -zz).^2)+(x(4)+1)-prange;
    A = [A dF];
end

k = l-F; % l is \ell (not one)
%k = -l+F;
N = A’*P;
c = N*k;
N = N*A;
deltahat = N\c;
% OLS solution
%deltahat = A\k;
% WLS-as-OLS solution
%sqrtP = sqrt(P);
%deltahat = (sqrtP*A)\(sqrtP*k)

khat = A*deltahat;
vhat = k-khat;

% prepare for iterations
x = x+deltahat;

% stop iterations
if max(abs(deltahat))<0.001
    break
end
%itertst = (k’*P*k)/(vhat’*P*vhat);
%if itertst < 1.000001
%    break
%end

end % iter --------------------------------------------------------------

% DOP
SSE = vhat’*P*vhat; %RSS or SSE
s02 = SSE/f; % MSE
s0 = sqrt(s02); %RMSE
Qdop = inv(A’*P*A);
Qx = s02.*Qdop;
Qdop = Qdop/sprior2;
PDOP = sqrt(trace(Qdop(1:3,1:3)));
% must be in local Easting-Northing-Up coordinates
%HDOP = sqrt(trace(Qdop(1:2,1:2)));
% must be in local Easting-Northing-Up coordinates
%VDOP = sqrt(Qdop(3,3));
TDOP = sqrt(Qdop(4,4));
GDOP = sqrt(trace(Qdop));

% Dispersion etc of elements
%Qx = s02.*inv(A’*P*A);
sigmas = sqrt(diag(Qx));
sigma = diag(sigmas);
isigma = inv(sigma);
% correlations between estimates
Rx = isigma*Qx*isigma;

% Standardised residuals
%Qv = s02.*(inv(P)-A*inv(A’*P*A)*A’);
Qv = s02.*inv(P)-A*Qx*A’;
sigmares = sqrt(diag(Qv));
stdres = vhat./sigmares;

disp(’--------------------------------------------------------’)
disp(’estimated parameters/elements [m]’)
x
disp(’estimated clock error [s]’)
x(4)/clight
disp(’number of iterations’)
iter
disp(’standard errors of elements [m]’)
sigmas
%tval = x./sigmas
disp(’s0’)
```

```
s0
disp('PDOP')
PDOP
%stdres
disp('difference between estimated elements and initial guess')
deltaori = x-x0
disp('difference between true values and estimated elements')
deltaori = xtrue-x
disp('-----------------------------------------------------------')

% t-values and probabilities of finding larger |t|
% pt should be smaller than, say, (5% or) 1%
t = x./sigmas;
pt = betainc(f./(f+t.^2),0.5*f,0.5);

% probabilitiy of finding larger s02
% should be greater than, say, 5% (or 1%)
pchi2 = 1-gammainc(0.5*SSE,0.5*f);

% semi-axes in confidence ellipsoid for position estimates
% 95% fractile for 3 dfs is  7.815 = 2.796^2
% 99% fractile for 3 dfs is 11.342 = 3.368^2
[vQx dQx] = eigsort(Qx(1:3,1:3));
semiaxes = sqrt(diag(dQx));
% 95% fractile for 2 dfs is  5.991 = 2.448^2
% 99% fractile for 2 dfs is  9.210 = 3.035^2

%   df    F(3,df).95  F(3,df).99
%    1      215.71      5403.1
%    2       19.164       99.166
%    3        9.277       29.456
%    4        6.591       16.694
%    5        5.409       12.060
%   10        3.708        6.552
%  100        2.696        3.984
%  inf        2.605        3.781
% chi^2 approximation, 95% fractile
figure
ellipsoidrot(0,0,0,semiaxes(1)*sqrt(7.815),semiaxes(2)*sqrt(7.815),...
    semiaxes(3)*sqrt(7.815),vQx);
axis equal
xlabel('X [m]'); ylabel('Y [m]'); zlabel('Z [m]');
title('95% confidence ellipsoid, ECEF, \chi^2 approx.')
% F approximation, 95% fractile. NB the fractile depends on df
figure
ellipsoidrot(0,0,0,semiaxes(1)*sqrt(3*9.277),semiaxes(2)*sqrt(3*9.277),...
    semiaxes(3)*sqrt(3*9.277),vQx);
axis equal
xlabel('X [m]'); ylabel('Y [m]'); zlabel('Z [m]');
title('95% confidence ellipsoid, ECEF, F approx.')
print -depsc2 confXYZ.eps
%% F approximation; number of obs goes to infinity
%figure
%ellipsoidrot(0,0,0,semiaxes(1)*sqrt(3*2.605),semiaxes(2)*sqrt(3*2.605),...
%    semiaxes(3)*sqrt(3*2.605),vQx);
%axis equal
%xlabel('X [m]'); ylabel('Y [m]'); zlabel('Z [m]');
%title('95% confidence ellipsoid, ECEF, F approx., nobs -> inf')

% To geographical coordinates, from Strang & Borre (1997)
[bb,ll,hh,phi,lambda] = c2gwgs84(x(1),x(2),x(3))

% Convert Qx (ECEF) to Qenu (ENU)
sp = sin(phi);
cp = cos(phi);
sl = sin(lambda);
cl = cos(lambda);
Ft = [-sl cl 0; -sp*cl -sp*sl cp; cp*cl cp*sl sp]; % ECEF -> ENU
Qenu = Ft*Qx(1:3,1:3)*Ft';
% std.err. of ENU
sigmasenu = sqrt(diag(Qenu));
[vQenu dQenu] = eigsort(Qenu(1:3,1:3));
semiaxes = sqrt(diag(dQenu));
```

```
% F approximation, 95% fractile. NB the fractile depends on df
figure
ellipsoidrot(0,0,0,semiaxes(1)*sqrt(3*9.277),semiaxes(2)*sqrt(3*9.277),...
    semiaxes(3)*sqrt(3*9.277),vQenu);
axis equal
xlabel('E [m]'); ylabel('N [m]'); zlabel('U [m]');
title('95% confidence ellipsoid, ENU, F approx.')
print -depsc2 confENU.eps
% Same thing, only more elegant
figure
ellipsoidrot(0,0,0,semiaxes(1)*sqrt(3*9.277),semiaxes(2)*sqrt(3*9.277),...
    semiaxes(3)*sqrt(3*9.277),Ft*vQx);
axis equal
xlabel('E [m]'); ylabel('N [m]'); zlabel('U [m]');
title('95% confidence ellipsoid, ENU, F approx.')

%PDOP = sqrt(trace(Qenu)/sprior2)/s0;
HDOP = sqrt(trace(Qenu(1:2,1:2))/sprior2)/s0;
VDOP = sqrt(Qenu(3,3)/sprior2)/s0;

% Studentized/jackknifed residuals
if f>1 studres = stdres./sqrt((f-stdres.^2)/(f-1));
end

function [bb,ll,h,phi,lambda] = c2gwgs84(x,y,z)

% C2GWGS84
%   Convertion of cartesian coordinates (X,Y,Z) to geographical
%   coordinates (phi,lambda,h) on the WGS 1984 reference ellipsoid
%
%   phi and lambda are output as vectors: [degrees minutes seconds]'

% Modified by Allan Aasbjerg Nielsen (2004) after
% Kai Borre 02-19-94
% Copyright (c) by Kai Borre
% $Revision: 1.0 $  $Date: 1997/10/15  %

a = 6378137;
f = 1/298.257223563;

lambda = atan2(y,x);
ex2 = (2-f)*f/((1-f)^2);
c = a*sqrt(1+ex2);
phi = atan(z/((sqrt(x^2+y^2)*(1-(2-f))*f)));

h = 0.1; oldh = 0;
while abs(h-oldh) > 1.e-12
   oldh = h;
   N = c/sqrt(1+ex2*cos(phi)^2);
   phi = atan(z/((sqrt(x^2+y^2)*(1-(2-f)*f*N/(N+h)))));
   h = sqrt(x^2+y^2)/cos(phi)-N;
end

phi1 = phi*180/pi;
b = zeros(1,3);
b(1) = fix(phi1);
b(2) = fix(rem(phi1,b(1))*60);
b(3) = (phi1-b(1)-b(2)/60)*3600;
bb = [b(1) b(2) b(3)]';
lambda1 = lambda*180/pi;
l = zeros(1,3);
l(1) = fix(lambda1);
l(2) = fix(rem(lambda1,l(1))*60);
l(3) = (lambda1-l(1)-l(2)/60)*3600;
ll = [l(1) l(2) l(3)]';
```

2.2 Nonlinear WLS by other Methods

The remaining sections describe a few other methods often used for solving the nonlinear (weighted) least squares regression problem.

2.2.1 The Gradient or Steepest Descent Method

Let us go back to Equation 181: $y_i = f_i(\boldsymbol{\theta}) + e_i$, $i = 1, \dots, n$ and consider the nonlinear WLS case

$$\|\boldsymbol{e}^2(\boldsymbol{\theta})\| = \boldsymbol{e}^T\boldsymbol{P}\boldsymbol{e}/2 = \frac{1}{2}\sum_{i=1}^{n} p_i[y_i - f_i(\boldsymbol{\theta})]^2. \tag{301}$$

The components of the gradient $\nabla\|\boldsymbol{e}^2\|$ are

$$\frac{\partial\|\boldsymbol{e}^2\|}{\partial\theta_k} = -\sum_{i=1}^{n} p_i[y_i - f_i(\boldsymbol{\theta})]\frac{\partial f_i(\boldsymbol{\theta})}{\partial\theta_k}. \tag{302}$$

From an initial value (an educated guess) we can update $\boldsymbol{\theta}$ by taking a step in the direction in which $\|\boldsymbol{e}^2\|$ decreases most rapidly, namely in the direction of the negative gradient

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta}_{old} - \alpha\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_{old})\| \tag{303}$$

where $\alpha > 0$ determines the step size. This is done iteratively until convergence.

2.2.2 Newton's Method

Let us now expand $\|\boldsymbol{e}^2(\boldsymbol{\theta})\|$ to second order around an initial value $\boldsymbol{\theta}_0$

$$\|\boldsymbol{e}^2(\boldsymbol{\theta})\| \simeq \|\boldsymbol{e}^2(\boldsymbol{\theta}_0)\| + [\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_0)\|]^T(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \frac{1}{2}[\boldsymbol{\theta} - \boldsymbol{\theta}_0]^T\boldsymbol{H}(\boldsymbol{\theta}_0)[\boldsymbol{\theta} - \boldsymbol{\theta}_0] \tag{304}$$

where $\boldsymbol{H} = \partial^2\|\boldsymbol{e}^2(\boldsymbol{\theta})\|/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T$ is the second order derivative of $\|\boldsymbol{e}^2(\boldsymbol{\theta})\|$ also known as the Hessian matrix not to be confused with the hat matrix in Equations 44, 117 and 209. The gradient of the above expansion is

$$\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta})\| \simeq \nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_0)\| + \boldsymbol{H}(\boldsymbol{\theta}_0)[\boldsymbol{\theta} - \boldsymbol{\theta}_0]. \tag{305}$$

At the minimum $\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta})\| = \boldsymbol{0}$ and therefore we can find that minimum by updating

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta}_{old} - \boldsymbol{H}_{old}^{-1}\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_{old})\| \tag{306}$$

until convergence.

From Equation 302 the elements of the Hessian $\boldsymbol{H}$ are

$$H_{kl} = \frac{\partial^2\|\boldsymbol{e}^2\|}{\partial\theta_k\partial\theta_l} = \sum_{i=1}^{n} p_i\left[\frac{\partial f_i(\boldsymbol{\theta})}{\partial\theta_k}\frac{\partial f_i(\boldsymbol{\theta})}{\partial\theta_l} - [y_i - f_i(\boldsymbol{\theta})]\frac{\partial^2 f_i(\boldsymbol{\theta})}{\partial\theta_k\partial\theta_l}\right]. \tag{307}$$

We see that the Hessian is symmetric. The second term in H_{kl} depends on the sum of the residuals between model and data, which is supposedly small both since our model is assumed to be good and since its terms can have opposite signs. It is therefore customary to omit this term. If the Hessian is positive definite we have a local minimizer and if its negative definite we have a local maximizer (if its indefinite, i.e., it has both positive and negative eigenvalues, we have a saddle point). $\boldsymbol{H}$ is sometimes termed the curvature matrix.

2.2.3 The Gauss-Newton Method

The basis of the Gauss-Newton method is a linear Taylor expansion of $\boldsymbol{e}$

$$\boldsymbol{e}(\boldsymbol{\theta}) \simeq \boldsymbol{e}(\boldsymbol{\theta}_0) + \boldsymbol{J}(\boldsymbol{\theta}_0)[\boldsymbol{\theta} - \boldsymbol{\theta}_0] \tag{308}$$

where $\boldsymbol{J}$ is the so-called Jacobian matrix containing the partial derivatives of $\boldsymbol{e}$ (like $\boldsymbol{A}$ containing the partial derivatives of $\boldsymbol{F}$ in Equation 192). In the WLS case this leads to

$$\frac{1}{2}\boldsymbol{e}^T(\boldsymbol{\theta})\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}) \simeq \frac{1}{2}\boldsymbol{e}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta})_0 + [\boldsymbol{\theta} - \boldsymbol{\theta}_0]^T\boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_0) + \frac{1}{2}[\boldsymbol{\theta} - \boldsymbol{\theta}_0]^T\boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_0)[\boldsymbol{\theta} - \boldsymbol{\theta}_0]. \tag{309}$$

The gradient of this expression is $\boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_0) + \boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_0)[\boldsymbol{\theta} - \boldsymbol{\theta}_0]$ and its Hessian is $\boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_0)$. The gradient evaluated at $\boldsymbol{\theta}_0$ is $\boldsymbol{J}^T(\boldsymbol{\theta}_0)\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_0)$. We see that the Hessian is independent of $\boldsymbol{\theta} - \boldsymbol{\theta}_0$, it is symmetric and it is positive definite if $\boldsymbol{J}(\boldsymbol{\theta}_0)$ is full rank corresponding to linearly independent columns. Since the Hessian is positive definite we have a minimizer and since $\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_{old})\| = \boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_{old})$ we get from Equation 306

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta}_{old} - [\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_{old})]^{-1}\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_{old}). \tag{310}$$

This corresponds to the normal equations for $\boldsymbol{\theta}_{new} - \boldsymbol{\theta}_{old}$

$$[\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_{old})](\boldsymbol{\theta}_{new} - \boldsymbol{\theta}_{old}) = -\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_{old}). \tag{311}$$

This is equivalent to Equation 194 so the linearization method described in Section 2.1 is actually the Gauss-Newton method with $-\boldsymbol{A}$ as the Jacobian.

It can be shown that if the function to be minimized is twice continuously differentiable in a neighbourhood around the solution $\boldsymbol{\theta}^*$, if $\boldsymbol{J}(\boldsymbol{\theta})$ over iterations is nonsingular, and if the initial solution $\boldsymbol{\theta}_0$, is close enough to $\boldsymbol{\theta}^*$, then the Gauss-Newton method converges. It can also be shown that the convergence is quadratic, i.e., the length of the increment vector ($\hat{\boldsymbol{\Delta}}$ in Section 2.1 and `h` in the Matlab function below) decreases quadratically over iterations.

Below is an example of a Matlab function implementation of the unweighted version of the Gauss-Newton algorithm. Note, that the code to solve the positioning problem is a `while` loop the body of which is three (or four) statements only. This is easily extended with the weighting and the statistics part given in the previous example (do this as an exercise). Note also, that in the call to function `gaussnewton` we need to call the function `fJ` with the at symbol (`@`) to create a Matlab function handle.

Matlab code for Example 11

```
function x = gaussnewton(fJ, x0, tol, itermax)

% x = gaussnewton(@fJ, x0, tol, itermax)
%
% gaussnewton solves a system of nonlinear equations
% by the Gauss-Newton method.
%
% fJ        - gives f(x) and the Jacobian J(x) by [f, J] = fJ(x)
% x0        - initial solution
% tol       - tolerance, iterate until maximum absolute value  of correction
%             is smaller than tol
% itermax   - maximum number of iterations
%
% x         - final solution
%
% fJ is written for the occasion

% (c) Copyright 2005
% Allan Aasbjerg Nielsen
% aa@imm.dtu.dk, www.imm.dtu.dk/~aa
% 8 Nov 2005

% Modified after
% L. Elden, L. Wittmeyer-Koch and H.B. Nielsen (2004).

if nargin < 2, error('too few input arguments'); end
if nargin < 3, tol = 1e-2; end
if nargin < 4, itermax = 100; end

iter = 0;
x = x0;
h = realmax*ones(size(x0));

while (max(abs(h)) > tol) && (iter < itermax)
    [f, J] = feval(fJ, x);
    h = J\f;
    x = x - h;
    iter = iter + 1;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [f, J] = fJ(x)
```

```
xxyyzz = [16577402.072   5640460.750 20151933.185;
          11793840.229 -10611621.371 21372809.480;
          20141014.004 -17040472.264  2512131.115;
          22622494.101  -4288365.463 13137555.567;
          12867750.433  15820032.908 16952442.746;
          -3189257.131 -17447568.373 20051400.790;
          -7437756.358  13957664.984 21692377.935]; % [m]
l = [20432524.0 21434024.4 24556171.0 21315100.2 21255217.0 24441547.2 ...
        23768678.3]'; % [m]

xx = xxyyzz(:,1);
yy = xxyyzz(:,2);
zz = xxyyzz(:,3);

range = sqrt((x(1)-xx).^2 + (x(2)-yy).^2 + (x(3)-zz).^2);
prange = range + x(4);
f = l - prange;

if nargout < 2, return; end

n = size(f,1); % # obs
p = 4; % # parameters
J = zeros(n,p);

% analytical derivatives
J(:,1) = -(x(1)-xx)./range;
J(:,2) = -(x(2)-yy)./range;
J(:,3) = -(x(3)-zz)./range;
J(:,4) = -ones(n,1);

return

% numerical derivatives
delta = 1;
for i = 1:p
    y = x;
    y(i) = x(i) + delta;
    J(:,i) = (fJ(y) - f); %./delta;
end

return

% or symmetrized
delta = 0.5;
for i = 1:p
    y = x;
    z = x;
    y(i) = x(i) + delta;
    z(i) = x(i) - delta;
    J(:,i) = (fJ(y) - fJ(z)); %./(2*delta);
end
```

2.2.4 The Levenberg-Marquardt Method

The Gauss-Newton method may cause the new $\boldsymbol{\theta}$ to wander off further from the minimum than the old $\boldsymbol{\theta}$ because of nonlinear components in $\boldsymbol{e}$ which are not modelled. Near the minimum the Gauss-Newton method converges very rapidly whereas the gradient method is slow because the gradient vanishes at the minimum. In the Levenberg-Marquardt method we modify Equation 311 to

$$[\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{J}(\boldsymbol{\theta}_{old}) + \mu\boldsymbol{I}](\boldsymbol{\theta}_{new} - \boldsymbol{\theta}_{old}) = -\boldsymbol{J}^T(\boldsymbol{\theta}_{old})\boldsymbol{P}\boldsymbol{e}(\boldsymbol{\theta}_{old}) \tag{312}$$

where $\mu \geq 0$ is termed the damping factor. The Levenberg-Marquardt method is a hybrid of the gradient method far from the minimum and the Gauss-Newton method near the minimum: if μ is large we step in the direction of the steepest descent, if $\mu = 0$ we have the Gauss-Newton method.

Also Newton's method may cause the new $\boldsymbol{\theta}$ to wander off further from the minimum than the old $\boldsymbol{\theta}$ since the Hessian may be indefinite or even negative definite (this is not the case for $\boldsymbol{J}^T\boldsymbol{P}\boldsymbol{J}$). In a Levenberg-Marquardt-like extension to Newton's method we could modify Equation 306 to

$$\boldsymbol{\theta}_{new} = \boldsymbol{\theta}_{old} - (\boldsymbol{H}_{old} + \mu\boldsymbol{I})^{-1}\nabla\|\boldsymbol{e}^2(\boldsymbol{\theta}_{old})\|. \tag{313}$$

3 Final Comments

In geodesy (and land surveying and GNSS) applications of regression analysis we are often interested in the estimates of the regression coefficients also known as the parameters or the elements which are often 2- or 3-D geographical positions, and their estimation accuracies. In many other application areas we are (also) interested in the ability of the model to predict values of the response variable from new values of the explanatory variables not used to build the model.

Unlike the Gauss-Newton method both the gradient method and Newton's method are general and not restricted to least squares problems, i.e., the functions to be optimized are not restricted to the form $\boldsymbol{e}^T\boldsymbol{e}$ or $\boldsymbol{e}^T\boldsymbol{P}\boldsymbol{e}$. Many other methods than the ones described and sketched here both general and least squares methods such as quasi-Newton methods, conjugate gradients and simplex search methods exist.

Solving the problem of finding a global optimum in general is very difficult. The methods described and sketched here (and many others) find a minimum that depends on the set of initial values chosen for the parameters to estimate. This minimum may be local. It is often wise to use several sets of initial values to check the robustness of the solution offered by the method chosen.

Literature

P.R. Bevington (1969). *Data Reduction and Error Analysis for the Physical Sciences.* McGraw-Hill.

K. Borre (1990). *Landmåling.* Institut for Samfundsudvikling og Planlægning, Aalborg. In Danish.

K. Borre (1992). *Mindste Kvadraters Princip Anvendt i Landmålingen.* Aalborg. In Danish.

M. Canty, A.A. Nielsen and M. Schmidt (2004). Automatic radiometric normalization of multitemporal satellite imagery. *Remote Sensing of Environment* **41**(1), 4-19.

P. Cederholm (2000). *Udjævning.* Aalborg Universitet. In Danish.

R.D. Cook and S. Weisberg (1982). *Residuals and Infuence in Regression.* Chapman & Hall.

K. Conradsen (1984). *En Introduktion til Statistik, vol. 1A-2B.* Informatik og Matematisk Modellering, Danmarks Tekniske Universitet. In Danish.

K. Dueholm, M. Laurentzius and A.B.O. Jensen (2005). *GPS. 3rd Edition.* Nyt Teknisk Forlag. In Danish.

L. Eldén, L. Wittmeyer-Koch and H.B. Nielsen (2004). *Introduction to Numerical Computation - analysis and MATLAB illustrations.* Studentlitteratur.

N. Gershenfeld (1999). *The Nature of Mathematical Modeling.*

Cambridge University Press.

G.H. Golub and C.F. van Loan (1996). *Matrix Computations, Third Edition.* Johns Hopkins University Press.

P.S. Hansen, M.P. Bendsøe and H.B. Nielsen (1987). *Lineær Algebra - Datamatorienteret.* Informatik og Matematisk Modellering, Matematisk Institut, Danmarks Tekniske Universitet. In Danish.

T. Hastie, R. Tibshirani and J. Friedman (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Second Edition.* Springer.

O. Jacobi (1977). *Landmåling 2. del. Hovedpunktsnet.* Den private Ingeniørfond, Danmarks Tekniske Universitet. In Danish.

A.B.O. Jensen (2002). *Numerical Weather Predictions for Network RTK.* Publication Series 4, volume 10. National Survey and Cadastre, Denmark.

N. Kousgaard (1986). *Anvendt Regressionsanalyse for Samfundsvidenskaberne.* Akademisk Forlag. In Danish.

K. Madsen, H.B. Nielsen and O. Tingleff (1999). *Methods for Non-Linear Least Squares Problems.* Informatics and Mathematical Modelling, Technical University of Denmark.

P. McCullagh and J. Nelder (1989). *Generalized Linear Models.* Chapman & Hall. London, U.K.

E.M. Mikhail, J.S. Bethel and J.C. McGlone (2001). *Introduction to Modern Photogrammetry.* John Wiley and Sons.

E. Mærsk-Møller and P. Frederiksen (1984). *Landmåling: Elementudjævning.* Den private Ingeniørfond, Danmarks Tekniske Universitet. In Danish.

A.A. Nielsen (2001). Spectral mixture analysis: linear and semi-parametric full and partial unmixing in multi- and hyperspectral image data. *International Journal of Computer Vision* **42**(1-2), 17-37 and *Journal of Mathematical Imaging and Vision* **15**(1-2), 17-37.

W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery (1992). *Numerical Recipes in C: The Art of Scientific Computing. Second Edition.* Cambridge University Press.

J.A. Rice (1995). *Mathematical Statistics and Data Analysis. Second Edition.* Duxbury Press.

G. Strang (1980). *Linear Algebra and its Applications. Second Edition.* Academic Press.

G. Strang and K. Borre (1997). *Linear Algebra, Geodesy, and GPS.* Wellesley-Cambridge Press.

P. Thyregod (1998). *En Introduktion til Statistik, vol. 3A-3D.* Informatik og Matematisk Modellering, Danmarks Tekniske Universitet. In Danish.

W.N. Venables and B.D. Ripley (1999). *Modern Applied Statistics with S-PLUS. Third Edition.* Springer.

Index