STATISTICS 578
REGRESSION ANALYSIS

Doug Wiens*

June 14, 2020

Contents

I Linear Regression 6

1 Introduction; Matrix formulation of regression model 7

2 LSEs; Gram-Schmidt Theorem and its consequences 19

3 Distributions; Confidence regions; LR test . 27

4 LR test in Normal models; acetylene data . 36

5 Ridge regression; Weighted and Generalized Least Squares 46

6 Logistic regression through WLS or Maximum likelihood 56

7 Influence measures 63

II Nonlinear Regression 71

8 Nonlinear models; Gauss-Newton algorithm 72

9 Likelihood regions 87

10 Lubricant data set; Starting values 96

11 Hypothesis testing 103

III Smoothing; Alternatives to Least Squares 112

12 Splines and other bases 113

13 Kernel Smoothing; Local Regression 121

14 Generalized additive modelling; Projection pursuit 133

15 Lasso; $n << p$; Quantile Regression 147

IV Robust Regression Methods 155

16 The need for robustness; M-estimation . . . 156

17 Huber's ψ_c; Computing M-estimates 164

18 Asymptotics; Inferences; Pseudovalues . . . 173

19 Breakdown and Influence 182

20 Generalized M-estimation; High breakdown estimates 190

21 One-step GM-estimation 198

22 MM-estimation 205

V Design 215

23 Classical regression designs 216

24 Robust regression designs I 227

25 Robust regression designs II 237

Part I

Linear Regression

1. Introduction; Matrix formulation of regression model

- Regression models – general framework

  observed random variable =
  function of 'covariates' + random error.

  That is, the experimenter wishes to obtain information about a r.v. Y, whose behaviour will presumably depend on the values of covariates $\mathbf{x} = (x_1, ..., x_p)'$. He/she sets the values of these (so, in particular, the covariates are *non-random*), and observes the resulting values of Y, apart from *random error* (e.g. measurement error). The 'function' $f(\boldsymbol{\theta}; \mathbf{x})$ referred to above typically has a known form, but may depend on unknown parameters $\boldsymbol{\theta}$.

- Example 1: In pharmacology and elsewhere the output (Y) of a chemical reaction may depend on the input x, random error ε and non-negative parameters θ_1, θ_2 according to a 'Michaelis-Menten' model
$$Y = \frac{\theta_1 x}{\theta_2 + x} + \varepsilon.$$
Note horizontal asymptote at θ_1, 'halfway point' is $x = \theta_2$. One observes pairs (Y_i, x_i) , $i = 1, ..., n$ and from these can estimate the parameters. Symbolically,
$$Y_i = f(\boldsymbol{\theta}; x_i) + \varepsilon_i, \; i = 1, ..., n.$$
The function $f(\boldsymbol{\theta}; x) = \frac{\theta_1 x}{\theta_2 + x}$ $(\boldsymbol{\theta} = (\theta_1, \theta_2)')$ is a non-linear function of $\boldsymbol{\theta}$; hence this is a *non-linear regression model*. With
$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \boldsymbol{\eta}(\boldsymbol{\theta}) = \begin{pmatrix} f(\boldsymbol{\theta}; x_1) \\ \vdots \\ f(\boldsymbol{\theta}; x_n) \end{pmatrix}, \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$
we have
$$\mathbf{Y} = \boldsymbol{\eta}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}. \tag{1.1}$$

- Typical assumptions on the errors:
  - Average error is zero: $E\left[\varepsilon_i\right] = 0$; thus
  $$E[\boldsymbol{\varepsilon}] = \begin{pmatrix} E[\varepsilon_1] \\ \vdots \\ E[\varepsilon_n] \end{pmatrix} = \mathbf{0} \text{ and so } E[\mathbf{Y}] = \boldsymbol{\eta}(\boldsymbol{\theta}).$$
  - Errors on different trials are uncorrelated, but all are equally varied: $\text{cov}\left[\varepsilon_i, \varepsilon_j\right] = 0$ if $i \neq j, = \sigma_\varepsilon^2$ if $i = j$; thus
  $$\begin{aligned} \text{cov}[\boldsymbol{\varepsilon}] &= E\left[(\boldsymbol{\varepsilon} - E[\boldsymbol{\varepsilon}])(\boldsymbol{\varepsilon} - E[\boldsymbol{\varepsilon}])'\right] \\ = & \; E\left[\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}'\right] = \begin{pmatrix} \text{cov}[\varepsilon_1, \varepsilon_1] & \cdots & \text{cov}[\varepsilon_1, \varepsilon_n] \\ \vdots & \ddots & \vdots \\ \text{cov}[\varepsilon_n, \varepsilon_1] & \cdots & \text{cov}[\varepsilon_n, \varepsilon_n] \end{pmatrix} \\ = & \begin{pmatrix} \sigma_\varepsilon^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_\varepsilon^2 \end{pmatrix} = \sigma_\varepsilon^2 \mathbf{I}_n \overset{why?}{=} \text{cov}[\mathbf{Y}]. \end{aligned}$$

- Example 2: Response (Y) to a drug depends on drug type ($x_1 = 0$ for control, $= 1$ for new drug) and amount administered (x_2). Possible model for response of i^{th} patient is

$$\begin{aligned} Y_i &= \theta_0 + \theta_1 x_{1,i} + \theta_2 x_{2,i} + \theta_{12} x_{1,i} x_{2,i} + \varepsilon_i, \text{ with} \\ E\left[Y_i\right] &= \begin{cases} \theta_0 + \theta_2 x_{2,i}, & \text{if control,} \\ (\theta_0 + \theta_1) + (\theta_2 + \theta_{12})\, x_{2,i}, & \text{if new drug.} \end{cases} \end{aligned}$$

Thus the difference in the mean responses at dose x_2 is $\theta_1 + \theta_{12} x_2$. A hypothesis of interest is then $H_0 : \theta_{12} = 0$; if true then the mean difference in responses is the same at all dosages.

- Example 2 in matrix terms: Let $\mathbf{Y}_{n \times 1}$ be the vector of responses from the n_0 patients on the control, followed by those from the n_1 patients on the new drug, then

$$\mathbf{Y} = \binom{\mathbf{1}_{n_0}}{\mathbf{1}_{n_1}} \theta_0 + \binom{\mathbf{0}_{n_0}}{\mathbf{1}_{n_1}} \theta_1 + \binom{\mathbf{x}_{2,0}}{\mathbf{x}_{2,1}} \theta_2 + \binom{\mathbf{0}_{n_0}}{\mathbf{x}_{2,1}} \theta_{12} + \boldsymbol{\varepsilon},$$

where $\mathbf{x}_{2,0}$ and $\mathbf{x}_{2,1}$ are the vectors of dosages and $\mathbf{0}$, $\mathbf{1}$ refer to vectors of zeroes and ones respectively. More succinctly,

$$\begin{aligned}\mathbf{Y} &= \begin{pmatrix} \mathbf{1}_{n_0} & \mathbf{0}_{n_0} & \mathbf{x}_{2,0} & \mathbf{0}_{n_0} \\ \mathbf{1}_{n_1} & \mathbf{1}_{n_1} & \mathbf{x}_{2,1} & \mathbf{x}_{2,1} \end{pmatrix} \boldsymbol{\theta} + \boldsymbol{\varepsilon} \\ &= \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}; \end{aligned} \tag{1.2}$$

here $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_{12})'$ and $\mathbf{X}$ is the $n \times 4$ 'design matrix'. Comparing (1.1) and (1.2),

$$E[\mathbf{Y}] = \boldsymbol{\eta}(\boldsymbol{\theta}) = \mathbf{X}\boldsymbol{\theta}$$

is a *linear* function of $\boldsymbol{\theta}$.

- Simple linear regression. $E[Y] = \theta_0 + \theta_1 x$, data $(x_i, y_i)_{i=1}^n$; $\mathbf{X}$ has columns $\mathbf{1}_n = (1, ..., 1)'$ and $(x_1, ..., x_n)'$.

- A <u>minimal</u> requirement for success in this course is that you NEVER write $\mathbf{X}^{-1}$ when $\mathbf{X}$ is a design matrix with, as is almost always the case, more rows than columns.

- Brief outline of this course:

  - Theory of linear models – canonical representation (this part is fairly theoretical); use this to quickly develop theory of estimation and hypothesis testing. Review some typical examples and applications.

  - Nonlinear regression – here the preceding theory is applied, with appropriate modifications, to treat nonlinear models. Relies on approximating a nonlinear response by a linear one; these approximations tend to be asymptotic in nature in that they become increasingly accurate as $n \to \infty$. The theory developed for linear models is also applied to give efficient *computational techniques*. (R package used extensively throughout the course.)

  - Robust regression – here we will study further modifications of those techniques developed for linear models; the purpose is to obtain procedures whose validity is maintained even when the assumptions underlying the model are violated (outlying observations, highly influential covariates, unsuspected correlations, etc.)

```
# Various regression techniques for the 'cars' data
# Y = braking distance, X = speed
# Data already in R; called 'cars'

x = cars$speed
y = cars$dist

par(mfrow=c(2,2))  # Sets the plotting function
to give a 2 by 2 panel of plots

## Two Least Squares fits
plot(x, y, xlab="speed", ylab="dist",
title(sub="LS fit"))

# Fit a straight line
fit1 = lm(dist ~speed, data = cars)
yhat = predict.lm(fit1)
```

```
lines(x,yhat)


# Fit a quadratic
fit2 = lm(dist ~speed + I(speed^2), data = cars)
 #omit the I() - what happens?
yhat = predict.lm(fit2)
lines(x,yhat)

legend(x=2, y=125, legend = paste
("lin.",1:2," = ", round(as.numeric(fit1$coef),2)))
legend(x=2, y=100, legend = paste
("quad.",1:3," = ", round(as.numeric(fit2$coef),2)))


## Two L1 - fits
## Here the sums of the ABSOLUTE VALUES
of the residuals(not their SQUARES) is minimized
## A special package for this has to be loaded:
# First go, in the menu, to Packages -> Set CRAN
mirror (to Canada (BC))
# Then Packages -> Load Packages -> quantreg
```

```
library(quantreg)

plot(x, y, xlab="speed", ylab="dist",
title(sub="L1 fit"))

# Fit a straight line
fit3 = rq(dist ~speed, data = cars)
yhat = predict(fit3)
lines(x,yhat)

# Fit a quadratic
fit4 = rq(dist ~speed + I(speed^2), data = cars)
yhat = predict(fit4)
lines(x,yhat)

legend(x=2, y=125, legend = paste
("lin.",1:2," = ", round(as.numeric(fit3$coef),2)))
legend(x=2, y=100, legend = paste
("quad.",1:3," = ", round(as.numeric(fit4$coef),2)))

## Here are two methods in which a model need not
## be specified:

## Smoothing spline fit (will be discussed later)
plot(x, y, xlab="speed", ylab="dist",
title(sub="spline fit"))

fit5 = smooth.spline(x,y)
yhat = predict(fit5, x=cars$speed)$y
lines(x, yhat)

## Loess fit (will be discussed later)
plot(x, y, xlab="speed", ylab="dist",
title(sub="loess fit"))

fit6 = loess(y˜x)
yhat = predict(fit6)
lines(x, yhat)
```

[natheight=7.1676in, natwidth=7.1676in,
height=6.4792in, width=6.4792in]
C:/sw50/temp/graphics/cars1$_{1.pdf}$
Various regression fits to the ‘cars’ data.

- Some theory for linear models. Here we consider more deeply the structure in (1.2), which implies that $E[\mathbf{Y}] = \mathbf{X}\boldsymbol{\theta}$. (You might first look at the STAT 512 notes, available on the web.) Denote by $\mathbf{z}_1, \cdots, \mathbf{z}_p \in \mathbb{R}^n$ the columns of $\mathbf{X}$, then

$$E[\mathbf{Y}] = \sum_{i=1}^{p} \mathbf{z}_i \theta_i$$

is a linear combination of the columns of $\mathbf{X}$. The set of all such linear combinations is a vector space, called the *column space* ($col(\mathbf{X})$), whose dimension is called the rank of $\mathbf{X}$.

- Two vectors $\mathbf{y}, \mathbf{z} \in \mathbb{R}^n$ are *orthogonal* if $\mathbf{y}'\mathbf{z} = \sum_{i=1}^{n} y_i z_i = 0$. We write $\mathbf{y} \perp \mathbf{z}$. The *Euclidean norm* (i.e., length) of $\mathbf{z}$ is $\|\mathbf{z}\| = \sqrt{\mathbf{z}'\mathbf{z}}$. The Least Squares Estimator (LSE) of $\boldsymbol{\theta}$ in a linear model is the vector $\hat{\boldsymbol{\theta}}$ which minimizes $\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|$. This is sometimes more conveniently expressed as the minimizer of

$$\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|^2 = \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i'\boldsymbol{\theta} \right)^2,$$

where $\mathbf{x}_1', ..., \mathbf{x}_n'$ are the *rows* of $\mathbf{X}$. In this notation $y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}$ is the i^{th} *residual.*

- Hat matrix: Consider a regression model $\mathbf{y} = \mathbf{X}\boldsymbol{\theta} + \varepsilon$ with $\mathbf{X}_{n\times p}$ of full rank p. We will later show that the LSEs are
$$\hat{\boldsymbol{\theta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y},$$
so that the estimate of $E\left[\mathbf{y}\right] = \mathbf{X}\boldsymbol{\theta}$ is $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\theta}} = \mathbf{H}\mathbf{y}$, where
$$\mathbf{H}_{n\times n} = \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'$$
is the 'hat' matrix – it 'places the hat on $\mathbf{y}$'.

  – Properties:
$$\begin{aligned} \mathbf{H} &= \mathbf{H}' \text{ ('symmetric')} \\ \mathbf{H}\mathbf{X} &= \mathbf{X} \\ (\mathbf{I}-\mathbf{H})\mathbf{X} &= \mathbf{0} \\ \mathbf{H}^2 &= \mathbf{H} \text{ ('idempotent')} \\ (\mathbf{I}-\mathbf{H})^2 &= (\mathbf{I}-\mathbf{H}) \\ \mathbf{H}(\mathbf{I}-\mathbf{H}) &= \mathbf{0}. \end{aligned}$$
  Also: $rk(\mathbf{H}) = p, rk(\mathbf{I}-\mathbf{H}) = n-p$.

2. LSEs; Gram-Schmidt Theorem and its consequences

- If $\mathbf{z}$ is any $n \times 1$ vector, and $\mathbf{H}$ is a hat matrix, then

$$\mathbf{z} = \mathbf{Hz} + (\mathbf{I} - \mathbf{H})\mathbf{z} = \mathbf{z}_1 + \mathbf{z}_2,$$

say, where $\mathbf{z}_1 \perp \mathbf{z}_2$. The first is in col($\mathbf{X}$) $=$ col$\,(\mathbf{H})$ (why?) and the second is in the space of vectors orthogonal to every vector in col($\mathbf{X}$). We write $\mathbf{z}_2 \in$ col$(\mathbf{X})^{\perp}$ ('orthogonal complement to col($\mathbf{X}$)'). You should verify that this is a vector space (i.e. is closed under addition and scalar multiplication), and that col$(\mathbf{X})^{\perp} =$ col$(\mathbf{I} - \mathbf{H})$, of dimension $n - p$.

- Least squares estimation in terms of hat matrix decomposition of norm of residuals: Note that $\mathbf{u} \perp \mathbf{v} \Rightarrow \|\mathbf{u} + \mathbf{v}\|^2 = \|\mathbf{u}\|^2 + \|\mathbf{v}\|^2$; then

$$\begin{aligned} \|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|^2 &= \|\mathbf{H}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})\|^2 + \|(\mathbf{I} - \mathbf{H})(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})\|^2 \\ &= \|\mathbf{H}(\mathbf{y} - \mathbf{X}\boldsymbol{\theta})\|^2 + \|(\mathbf{I} - \mathbf{H})\mathbf{y}\|^2 . \end{aligned}$$

The second (non-negative) term above does not depend on $\boldsymbol{\theta}$. If we can choose $\boldsymbol{\theta}$ so that the first term vanishes, we will have minimized $\|\mathbf{y}-\mathbf{X}\boldsymbol{\theta}\|^2$, and found that the minimum value is $\|(\mathbf{I}-\mathbf{H})\,\mathbf{y}\|^2$. We have $\mathbf{H}(\mathbf{y}-\mathbf{X}\boldsymbol{\theta})=0 \Leftrightarrow \mathbf{Hy}=\mathbf{X}\boldsymbol{\theta} \Leftrightarrow$

$$\boldsymbol{\theta}=\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}. \qquad (2.1)$$

Reason: the '$\Leftarrow$' is obvious; in the other direction we have that $\mathbf{Hy}=\mathbf{X}\boldsymbol{\theta} \Rightarrow \mathbf{X}'\mathbf{Hy}=\mathbf{X}'\mathbf{X}\boldsymbol{\theta}$, i.e.

$$\mathbf{X}'\mathbf{y}=\mathbf{X}'\mathbf{X}\boldsymbol{\theta}.$$

These are the 'normal equations', and (2.1) follows and gives the LS estimator $\hat{\boldsymbol{\theta}}$. The *fitted values* are

$$\hat{\mathbf{y}}=\mathbf{X}\hat{\boldsymbol{\theta}}=\mathbf{Hy},$$

and are orthogonal to the *residuals*

$$\mathbf{e}=\mathbf{y}-\hat{\mathbf{y}}=(\mathbf{I}-\mathbf{H})\,\mathbf{y}.$$

We say that $\mathbf{H}$ and $\mathbf{I}-\mathbf{H}$ *project* the data $(\mathbf{y})$ onto the estimation space and error space, respectively, and that these spaces are *orthogonal*.

- A square matrix $\mathbf{Q}_{n\times n}$ is *orthogonal* if the columns are mutually orthogonal, and have unit norm. Equivalently
$$\mathbf{QQ}' = \mathbf{Q}'\mathbf{Q} = \mathbf{I}_n.$$
If $\mathbf{Q}$ is orthogonal then $\|\mathbf{Qy}\| = \|\mathbf{y}\|$ for any $n\times 1$ vector $\mathbf{y}$ – 'norms are preserved'. Similarly, angles between vectors are also preserved (why?). Geometrically, an orthogonal transformation is a 'rigid motion' – it corresponds to a rotation and/or an interchange of two or more axes. It is possible to find a basis for col($\mathbf{X}$) consisting of mutually orthonormal vectors; this makes both the theory and the computations much simpler.

- **Gram-Schmidt Theorem**: Every m-dimensional vector space V, with basis $\{\mathbf{v}_1, ...\mathbf{v}_m\}$ say, has an orthonormal basis $\{\mathbf{q}_1, ...\mathbf{q}_m\}$. This basis can be constructed in such a way that $\mathbf{q}_j$ is a linear combination of $\left\{\mathbf{v}_1, ...\mathbf{v}_j\right\}$ only (and the coefficient of $\mathbf{v}_j$ is positive).

  Proof: Stat 512 notes.

- **QR-decomposition**. If $\mathbf{V}_{n\times m}$ has rank m, so that its columns $\{\mathbf{v}_1, ...\mathbf{v}_m\}$ are independent, hence form a basis of col(V), then we can apply Gram-Schmidt so has to get a matrix $\mathbf{Q}_{n\times m}$, whose columns $\{\mathbf{q}_1, ...\mathbf{q}_m\}$ are orthonormal. Since $\mathbf{q}_j$ was obtained as a linear combination of $\mathbf{v}_1, ...\mathbf{v}_j$, we can write

  $$\mathbf{V}_{n\times m}\mathbf{U}_{m\times m} = \mathbf{Q}_{n\times m},$$

  for $\mathbf{U}$ upper triangular with positive diagonal elements. Then $\mathbf{U}$ is nonsingular and $\mathbf{V} = \mathbf{QR}$ for $\mathbf{R} = \mathbf{U}^{-1}$. (Note that $\mathbf{R}$ is also upper triangular with positive diagonal elements.)

- We apply the decomposition arising from the Gram-Schmidt Theorem to regression, assuming that the design matrix $\mathbf{X}_{n\times p}$ has rank p. Write $\mathbf{X} = \mathbf{Q}_1\mathbf{R}_1$, where $\mathbf{Q}_1$: $n\times p$ has orthonormal columns, and $\mathbf{R}_1$: $p \times p$ is upper triangular with positive diagonal elements. Apply Gram-Schmidt once again, starting with the $n - p$ independent columns of $\mathbf{I} - \mathbf{H}$, to obtain $\mathbf{Q}_2$: $n \times (n - p)$

whose columns are orthonormal and are a basis for $col(\mathbf{X})^{\perp}$. Then $\mathbf{Q} \overset{def.}{=} (\mathbf{Q}_1 \vdots \mathbf{Q}_2)$ has orthonormal columns and is square, hence is an orthogonal matrix. We have

$$\begin{aligned}
\mathbf{QR} &= (\mathbf{Q}_1 \vdots \mathbf{Q}_2)\begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{pmatrix} = \mathbf{X}, \\
\mathbf{R}'\mathbf{R} &= \mathbf{R}_1'\mathbf{R}_1 = \mathbf{X}'\mathbf{X}, \\
\left(\mathbf{X}'\mathbf{X}\right)^{-1} &= \mathbf{R}_1^{-1}\mathbf{R}_1^{-1'}, \\
\mathbf{H} &= \mathbf{Q}_1\mathbf{Q}_1', \\
\mathbf{I} - \mathbf{H} &= \mathbf{Q}_2\mathbf{Q}_2'.
\end{aligned}$$

- In terms of QR-decomposition: we have that $\hat{\boldsymbol{\theta}} = \mathbf{R}_1^{-1}\mathbf{R}_1^{-1'}\mathbf{R}_1'\mathbf{Q}_1'\mathbf{y}$; i.e.

  $$\mathbf{R}_1\hat{\boldsymbol{\theta}} = \mathbf{Q}_1'\mathbf{y}.$$

  Thus compute

  $$\begin{aligned}
  \mathbf{z}_{n\times 1} &= \mathbf{Q}'\mathbf{y} = \begin{pmatrix} \mathbf{Q}_1' \\ \mathbf{Q}_2' \end{pmatrix} \mathbf{y} \\
  &= \begin{pmatrix} \mathbf{Q}_1'\mathbf{y} \\ \mathbf{Q}_2'\mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{pmatrix} \begin{matrix} p \times 1 \\ (n-p) \times 1 \end{matrix} .
  \end{aligned}$$

Then backsolve the system of equations $\mathbf{R}_1\hat{\boldsymbol{\theta}} = \mathbf{z}_1$. Numerically stable – no matrix inversions. It is done this way on R.

- The residual vector is $\mathbf{e} = \mathbf{Q}_2\mathbf{z}_2$, with squared norm $\|\mathbf{z}_2\|^2$. The usual estimate of the variance σ_ε^2 of the random errors ε is

$$S^2 = \frac{\text{SS of residuals}}{n-p} = \frac{\|\mathbf{e}\|^2}{n-p} = \frac{\|\mathbf{z}_2\|^2}{n-p},$$

the *mean squared error*. Commonly, the SS of residuals is called SSE (SS of Errors) and $S^2 = SSE/(n-p)$ is called MSE (Mean Squared Error). We have

$$E[\mathbf{z}] = \mathbf{Q}'E[\mathbf{y}] = \begin{pmatrix} \mathbf{Q}_1'\mathbf{Q}_1\mathbf{R}_1\boldsymbol{\theta} \\ \mathbf{Q}_2'\mathbf{Q}_1\mathbf{R}_1\boldsymbol{\theta} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_1\boldsymbol{\theta} \\ \mathbf{0} \end{pmatrix},$$

and then using the general result 'cov$[\mathbf{Ay}] = \mathbf{A}\text{cov}[\mathbf{y}]\mathbf{A}'$' (how?) we get

$$\text{cov}[\mathbf{z}] = \mathbf{Q}'COV[\mathbf{y}]\mathbf{Q} = \mathbf{Q}'\sigma^2\mathbf{I}\mathbf{Q} = \sigma^2\mathbf{I};$$

hence the elements $z_{p+1}, \ldots, z_n$ of $\mathbf{z}_2$ have mean zero and var$[z_i] = E\left[z_i^2\right] = \sigma^2$. Thus $S^2 = MSE$ is unbiased:

$$E\left[S^2\right] = E\left[\frac{\sum_{p+1}^{n} z_i^2}{n-p}\right] = \sigma_\varepsilon^2.$$

- **Maximum Likelihood.** So far none of this has required any assumptions about the probability distribution of the random errors. In addition to the assumptions that these be uncorrelated, mean zero, equally varied, assume now that they are normally distributed:

  $$\varepsilon_1, \ldots, \varepsilon_n \overset{i.i.d.}{\sim} N(0, \sigma_\varepsilon^2).$$

  Then $Y_i \sim N(\mathbf{x}_i'\boldsymbol{\theta}, \sigma_\varepsilon^2)$ and the Y_i are *independent* (rather than merely uncorrelated):

  $$\mathbf{Y} \sim N\left(\mathbf{X}\boldsymbol{\theta}, \sigma_\varepsilon^2\mathbf{I}_n\right).$$

  The *likelihood function* (= p.d.f. of the data, evaluated at the observed values) is

$$\begin{aligned} L\left(\boldsymbol{\theta},\sigma_\varepsilon^2\right) &= \prod_{i=1}^n \left(2\pi\sigma_\varepsilon^2\right)^{-1/2} e^{-\frac{\left(y_i-\mathrm{x}_i'\boldsymbol{\theta}\right)^2}{2\sigma_\varepsilon^2}} \\ &= \left(2\pi\sigma_\varepsilon^2\right)^{-n/2} e^{-\frac{\|\mathbf{y}-\mathbf{X}\boldsymbol{\theta}\|^2}{2\sigma_\varepsilon^2}}. \end{aligned}$$

The *maximum likelihood estimates* are the maximizers of L, or equivalently of the log-likelihood

$$l\left(\boldsymbol{\theta},\sigma_\varepsilon^2\right) = \log L\left(\boldsymbol{\theta},\sigma_\varepsilon^2\right) = -\frac{n}{2}\log\sigma_\varepsilon^2 - \frac{S\left(\boldsymbol{\theta}\right)}{2\sigma_\varepsilon^2} + const.,$$

where $S\left(\boldsymbol{\theta}\right) = \|\mathbf{y}-\mathbf{X}\boldsymbol{\theta}\|^2$ and 'const.' $= -\frac{n}{2}\log 2\pi$. The maximizing $\boldsymbol{\theta}$ is, clearly, the minimizer of $S\left(\boldsymbol{\theta}\right)$; quite generally in normal models **MLE = LSE**. Then solving

$$\frac{d}{d\sigma_\varepsilon^2}\left\{-\frac{n}{2}\log\sigma_\varepsilon^2 - \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{2\sigma_\varepsilon^2}\right\} = 0$$

results in

$$\hat{\sigma}^2_{MLE} = \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{n} = \frac{n-p}{n}S^2.$$

Again typically, the MLE of the variance is biased but the bias is easily removable.

3. Distributions; Confidence regions; LR test

- The derivations of the distributions of related quantities rely on several important properties of normally distributed r.v.s:

1. If $\mathbf{Z}_{r\times 1} \sim N_r\left(\mu, \mathbf{\Sigma}\right)$ and $\mathbf{A}_{q\times r}$ is a matrix of constants then $\mathbf{AZ}$ is normally distributed:
$$\mathbf{AZ} \sim N_q\left(\mathbf{A}\mu, \mathbf{A\Sigma A}'\right).$$

2. If $\mathbf{Z} \sim N\left(0, \sigma_\varepsilon^2 \mathbf{I}_r\right)$ then $\left\|\mathbf{Z}\right\|^2/\sigma_\varepsilon^2 \sim \chi_r^2$.

3. If $Z \sim N\left(\mu, \sigma^2\right)$ and $rS^2 \sim \sigma^2\chi_r^2$ independently of Z, then $\left(Z-\mu\right)/S \sim t_r$.

4. If $r_1 S_1^2 \sim \sigma^2\chi_{r_1}^2$, independently of $r_2 S_2^2 \sim \sigma^2\chi_{r_2}^2$, then $S_1^2/S_2^2 \sim F_{r_2}^{r_1}$.

- Suppose $\mathbf{y} \sim N\left(\mathbf{X}\boldsymbol{\theta}, \sigma_\varepsilon^2 \mathbf{I}_n\right)$. Put $\xi = \mathbf{X}\boldsymbol{\theta}$. The 'fitted values' are $\hat{\xi} = \mathbf{X}\hat{\boldsymbol{\theta}}$ (previously, and more commonly, written $\hat{\mathbf{y}}$.) From the QR-decomposition we obtained (p. 24)

$$\begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \end{pmatrix} \overset{def}{=} \begin{pmatrix} \mathbf{Q}_1^{'}\mathbf{y} \\ \mathbf{Q}_2^{'}\mathbf{y} \end{pmatrix} \sim N\left(\begin{pmatrix} \mathbf{R}_1\boldsymbol{\theta} \\ \mathbf{0} \end{pmatrix}, \sigma_\varepsilon^2 \mathbf{I}_n\right)$$

and

$$\begin{aligned} \hat{\xi} &= \mathbf{H}\mathbf{y} = \mathbf{Q}_1\mathbf{Q}_1^{'}\mathbf{y} = \mathbf{Q}_1\mathbf{z}_1, \\ \hat{\boldsymbol{\theta}} &= \mathbf{R}_1^{-1}\mathbf{z}_1, \\ S^2 &= \frac{\left\|\mathbf{z}_2\right\|^2}{n-p}. \end{aligned}$$

Thus:

5. $\hat{\xi}$ and $\hat{\boldsymbol{\theta}}$ are independent of S^2;

6. $S^2/\sigma_\varepsilon^2 \sim \chi_{n-p}^2/\left(n-p\right)$;

7. $\hat{\boldsymbol{\theta}} \sim N\left(\boldsymbol{\theta}, \sigma_\varepsilon^2 \mathbf{R}_1^{-1}\mathbf{R}_1^{-1\prime} = \sigma_\varepsilon^2 \left(\mathbf{X}'\mathbf{X}\right)^{-1}\right)$;

8. $\hat{\xi} \sim N\left(\xi, \sigma_\varepsilon^2 \mathbf{H}\right)$.

- From 7, $\hat{\theta}_j \sim N\left(\theta_j, \sigma_\varepsilon^2 \left[(\mathbf{X}'\mathbf{X})^{-1}\right]_{jj}\right)$. The standard error (= est'd. s.d. of an estimate) is $s\left(\hat{\theta}_j\right) = S \cdot \left(\left[(\mathbf{X}'\mathbf{X})^{-1}\right]_{jj}\right)^{1/2}$; then using 3,

  $$\frac{\hat{\theta}_j - \theta_j}{s\left(\hat{\theta}_j\right)} \sim t_{n-p}.$$

  (Used for marginal hypothesis tests and confidence intervals.)

- From 7 again, $\mathbf{R}_1\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \sim N\left(0, \sigma_\varepsilon^2 \mathbf{I}_p\right)$; then by 2,

  $$\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)' \mathbf{X}'\mathbf{X}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) = \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)' \mathbf{R}_1' \mathbf{R}_1 \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)$$

  $\sim \sigma_\varepsilon^2 \chi_p^2$, ind. of $S^2 \sim \sigma_\varepsilon^2 \chi_{n-p}^2 / (n-p)$. It follows that

  $$\frac{\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)' \mathbf{X}'\mathbf{X}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)}{pS^2} \sim F_{n-p}^p.$$

  Thus a $100(1-\alpha)\,\%$ confidence region for $\boldsymbol{\theta}$ is the ellipsoid

  $$\left\{\boldsymbol{\theta} \mid \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)' \mathbf{X}'\mathbf{X}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \leq pS^2 F_{n-p}^p (1-\alpha)\right\}.$$

- The expected value of Y at level $\mathbf{x}_0$ is $E\left[Y|\mathbf{x}_0\right] = \mathbf{x}_0^{'}\boldsymbol{\theta}$, with estimate

$$\mathbf{x}_0'\hat{\boldsymbol{\theta}} \sim N\left(\mathbf{x}_0'\boldsymbol{\theta}, \sigma_\varepsilon^2\mathbf{x}_0'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_0\right).$$

  It follows that a $100\left(1-\alpha\right)\%$ confidence interval is $\mathbf{x}_0'\hat{\boldsymbol{\theta}} \pm S\sqrt{\mathbf{x}_0'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_0} \cdot t_{n-p}\left(1-\frac{\alpha}{2}\right)$.

- Simultaneous confidence intervals on *all* $\mathbf{x}_0'\boldsymbol{\theta}$ are given by

$$\mathbf{x}_0'\hat{\boldsymbol{\theta}} \pm S\sqrt{\mathbf{x}_0'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_0} \cdot \sqrt{pF_{n-p}^{p}\left(1-\alpha\right)};$$

  the interpretation is that, before sampling, the probability is $1-\alpha$ that these intervals will contain $\mathbf{x}_0'\boldsymbol{\theta}$ for *every* $\mathbf{x}_0$. (Exercise.)

- Hypothesis testing in linear models. In the most general formulation of linear hypotheses, we have a 'full' model that specifies that ξ (the mean vector of $\mathbf{Y}$) lies in a particular vector space Π $(=$ col$\left(\mathbf{X}\right))$. We wish to test the hypothesis

H that ξ lies in a subspace Π_0 of Π (the 'restricted' model). The alternate hypothesis K is that $\xi \notin \Pi_0$.

- Example 1: Suppose that the p columns of $\mathbf{X}$ are independent, and that the first q columns form a basis for Π_0. Partition the vector of regression parameters as

  $$\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{\theta}_1 \\ \boldsymbol{\theta}_2 \end{pmatrix} \begin{matrix} q \\ p-q \end{matrix}.$$

  Then the null hypothesis is $H: \ \boldsymbol{\theta}_2 = \mathbf{0}$. (As in Example 2, Lecture 1.) We can always choose $\mathbf{X}$ (via Gram-Schmidt) so as to arrange matters in this way.

– Example 2: Testing Lack of Fit. Here we test if our postulated regression model is appropriate. The method requires that we have *replicates*, viz. there are $r < n$ distinct values $\mathbf{x}_1, ..., \mathbf{x}_r$ of $\mathbf{x}$, with n_i observations $\left\{Y_{ij}\right\}_{j=1}^{n_i}$ made at $\mathbf{x}_i$ ($\sum_{i=1}^{r} n_i = n$). The model specifies no particular regression structure: $E\left[Y_{ij}\right] = \mu_i$, with no necessary relationship among the μ_i. Thus

$$\xi = \begin{pmatrix} \mu_1 \mathbf{1}_{n_1} \\ \vdots \\ \mu_r \mathbf{1}_{n_r} \end{pmatrix} = \sum_{i=1}^{r} \begin{pmatrix} \mathbf{0}_{n_1} \\ \vdots \\ \mathbf{1}_{n_i} \\ \vdots \\ \mathbf{0}_{n_r} \end{pmatrix} \mu_i$$

and Π is the vector space consisting of all such ξ as the μ_i range over $\mathbb{R}$. This space has dimension r. The hypothesis H is that $\mu_i = \mathbf{x}_i' \boldsymbol{\theta}$ for some $\boldsymbol{\theta} \in \mathbb{R}^p$ and $p < r$; thus Π_0 is a p-dimensional subspace of Π (assuming that $\mathbf{x}_1, ..., \mathbf{x}_r$ are linearly independent; if so then what is a basis?).

- Maximum likelihood is the most widely applied method of estimation; its analogue in testing is the 'likelihood ratio test'. The idea is to compare the maximized likelihood under the union $H \cup K$ of the null and alternate hypotheses (the full model) with that under H alone (the reduced model); if the former is significantly larger than the latter we conclude that H is more restrictive than is justified by the data and 'reject' it.

- For a linear regression model this requires the computation of

$$\lambda = \frac{\max_{H \cup K} L\left(\boldsymbol{\theta}, \sigma_\varepsilon^2\right)}{\max_H L\left(\boldsymbol{\theta}, \sigma_\varepsilon^2\right)}.$$

In general (i.e. without assuming normality), if Π has dimension p and Π_0 has dimension q, then under H,

$$\begin{aligned} 2\log\lambda &= 2\left(\max_{H \cup K} l\left(\boldsymbol{\theta}, \sigma_\varepsilon^2\right) - \max_H l\left(\boldsymbol{\theta}, \sigma_\varepsilon^2\right)\right) \\ &\quad \xrightarrow{d} \chi^2_{p-q} \text{ as } n \to \infty. \end{aligned}$$

An asymptotically valid level α test then rejects if $2\log\lambda > \chi^2_{p-q}(1-\alpha)$.

- For Normal models we can evaluate λ more explicitly, and obtain exact distributional results. Recall

$$\begin{aligned} l\left(\boldsymbol{\theta}, \sigma^2_\varepsilon\right) &= -\frac{n}{2}\log\sigma^2_\varepsilon - \frac{S(\boldsymbol{\theta})}{2\sigma^2_\varepsilon} - \frac{n}{2}\log 2\pi; \text{ hence} \\ \max_{\sigma^2_\varepsilon} l\left(\boldsymbol{\theta}, \sigma^2_\varepsilon\right) &= l\left(\boldsymbol{\theta}, \frac{S(\boldsymbol{\theta})}{n}\right) \\ &= -\frac{n}{2}\log\frac{S(\boldsymbol{\theta})}{n} - \frac{n}{2}\left(1+\log 2\pi\right). \end{aligned}$$

Let $\hat{\boldsymbol{\theta}}$ be the minimizer of $S(\boldsymbol{\theta})$ under $H \cup K$ (the 'unrestricted MLE') and let $\hat{\hat{\boldsymbol{\theta}}}$ be the minimizer under H (the 'restricted MLE'). We write $S\left(\hat{\boldsymbol{\theta}}\right) = SSE_{Full}$, $S\left(\hat{\hat{\boldsymbol{\theta}}}\right) = SSE_{Red}$ and then

$$\begin{aligned} 2\log\lambda &= 2\left(-\frac{n}{2}\log\frac{SSE_{Full}}{n} + \frac{n}{2}\log\frac{SSE_{Red}}{n}\right) \\ &= n\log\frac{SSE_{Red}}{SSE_{Full}}. \end{aligned}$$

Thus H is rejected for large values of SSE_{Red}/SSE_{Full}. Equivalently, reject for large

$$\begin{aligned} F &= \left\{ \frac{SSE_{Red}}{SSE_{Full}} - 1 \right\} \frac{n-p}{p-q} = \frac{\frac{SSE_{Red} - SSE_{Full}}{p-q}}{\frac{SSE_{Full}}{n-p}} \\ &= \frac{\frac{\text{increase in the minimized } SS \text{ resulting from } H}{\text{change in d.f.}}}{\frac{\text{absolute minimum } SS}{\text{d.f. in full model}}}. \end{aligned}$$

- Here the 'minimized SS' is the sum of squares of the residuals, in the model (full or restricted) being considered. In Assignment 1 Q3 you are showing, in the general context of hypothesis testing considered above, that the F is distributed as $F \sim F^{p-q}_{n-p}$ <u>when the errors are normally distributed</u> and <u>H is true</u>.

4. LR test in Normal models; acetylene data

- To compute F typically requires us to run two regressions, one without H and the other assuming H; the relevant sums of squares are then read off of the printout. When H is true, $F \sim F_{n-p}^{p-q}$ and so the p-value is $P\left(F_{n-p}^{p-q} > F_{obs}\right)$ (assigned). Note also that typically the numerator d.f. $(p-q)$ is the reduction in the number of regression parameters when H is assumed.

- Testing LOF. Under $H \cup K$, the SS is

$$SS = \sum_{i=1}^{r} \sum_{j=1}^{n_i} \left(y_{ij} - \mu_i\right)^2$$

and is minimized by $\hat{\mu}_i =$ what?; hence

$$SSE_{Full} = \min_{H \cup K} SS = \sum_{i=1}^{r} \left(n_i - 1\right) S_i^2,$$

where $S_i^2 = \ldots$. One often writes $SSE_{Full} = SSPE$ (SS due to ‘pure error’); it is on $\sum_{i=1}^{r} \left(n_i - 1\right) =$

$n{-}r$ d.f. Under H, the minimum SS is $SSE_{Red} = SSE$ from the regression output with design matrix with rows $\mathbf{x}_i'$ repeated n_i times; it is on $n-p$ d.f. Then

$$F = \frac{\frac{SSE-SSPE}{r-p}}{\frac{SSPE}{n-r}}.$$

One often writes $SSE - SSPE$ as $SSLOF$ and then $F = MSLOF/MSPE$. One refers to the F_{n-r}^{r-p} distribution for the p-value. Note that $SSPE$ can be obtained from the output of an ANOVA (where it will appear as the SS of the residuals in a call such as `aov(y~as.factor(x))`).

- Why is the F an F? And what if the hypothesis is false? In Assignment 1 Q3 you are showing, in the general context of hypothesis testing with Normal errors, that

$$F \sim \frac{\|\mathbf{z}_2\|^2/(p-q)}{\|\mathbf{z}_3\|^2/(n-p)},$$

where $\mathbf{z}_2$ and $\mathbf{z}_3$ are Normal and <u>independent</u>, $\mathbf{z}_3 \sim N\left(0, \sigma^2 \mathbf{I}_{n-p}\right)$ (so $\|\mathbf{z}_3\|^2/(n-p) \sim \sigma^2\chi^2_{n-p}$) but $\mathbf{z}_2 \sim N\left(\eta, \sigma^2\mathbf{I}_{p-q}\right)$. When H is true, $\eta = 0$ and so $\|\mathbf{z}_2\|^2/(p-q) \sim \sigma^2\chi^2_{p-q}$ and $F \sim F^{p-q}_{n-p}$. But when the hypothesis is false, the distribution depends on η through $\delta^2 = \|\eta\|^2/\sigma^2$ and $\|\mathbf{z}_2\|^2 \sim \sigma^2 \sum_{k=1}^{p-q} Z_k^2$, where the Z_k are independent and Normal, but not with zero means: $\|\eta\|^2$ is the SS of their means. In this case we say that $\|\mathbf{z}_2\|^2/(p-q) \sim \sigma^2\chi^2_{p-q}\left(\delta^2\right)$, the non-central χ^2_{p-q} with 'non-centrality parameter' δ^2, and F is $\sim F^{p-q}_{n-p}\left(\delta^2\right)$, the non-central F. In the assignment you are finding a way to work out δ^2 explicitly as the squared distance between the true mean ξ and the closest vector in Π_0. So δ^2 increases as the null hypothesis becomes 'less true'. From these representations it is an easy matter to show that the 'power'

$$P\left(F^{p-q}_{n-p}\left(\delta^2\right) > \text{ critical value}\right)$$

is an increasing function of δ^2 - an intuitively pleasing property.

- Another hypothesis testing example:

```
> #R example; acetylene data
> # Data from Montgomery & Peck Example 8.1
> # Response variable (conv) is % conversion
 of n-heptane to acetylene
> # Explanatory variables temp(reactor temp),
 mole (chemical ratio),cont (contact time)
> # Enter the data:
> conv =c(49,50.2,50.5,...)
> temp = c(rep(1300,6),rep(1200,6),...)
> mole = c(7.5,9,11,13.5,...)
> cont = c(120,120,115,...)/10000
>
> # Put the data into a "frame"; look at all
pairs of plots
> acet = data.frame(conv, temp, mole, cont)
> pairs(acet) # Note that the predictors cont
and temp are highly correlated
```

[natheight=7.1676in, natwidth=7.1676in,
height=4.8888in, width=4.8888in]
C:/sw50/temp/graphics/acet1$_{fig}1_{2.pdf}$
All pairs of variables from “acet” dataframe.

```
> # Fit a full second order model
> x = cbind(temp, mole, cont, temp*mole, temp*cont,
mole*cont, temp^2, mole^2, cont^2)
> dimnames(x) = list(NULL, c("T", "M", "C",
"T*M", "T*C", "M*C", "T2", "M2", "C2"))
> fit1 = lsfit(x, conv); ls.print(fit1)
Residual Standard Error=0.9014
R-Square=0.9977
F-statistic (df=9, 6)=289.7
p-value=0

            Estimate   Std.Err t-value Pr(>|t|)
Intercept -3.617e+03 3.136e+03  -1.153   0.2926
T          5.324e+00 4.880e+00   1.091   0.3171
M          1.924e+01 4.303e+00   4.473   0.0042
C          1.377e+04 1.045e+04   1.318   0.2357
T*M       -1.410e-02 3.200e-03  -4.404   0.0045
T*C       -1.058e+01 8.241e+00  -1.284   0.2467
M*C       -2.103e+01 9.241e+00  -2.276   0.0631
T2        -1.900e-03 1.900e-03  -1.016   0.3487
M2        -3.030e-02 1.170e-02  -2.597   0.0408
C2        -1.158e+04 7.699e+03  -1.504   0.1832
```

```
> d = ls.diag(fit1); print(d)
$std.dev   # This is ’S’
[1] 0.9014

$hat  # Diagonal elements of hat matrix
 [1] 0.5295 0.3060 0.4007 ...

$std.res  # Residuals divided by their
              std. deviations
 [1] -0.9920  0.5753  0.7084 ...

$stud.res  # Later
 [1] -0.9904  0.5403  0.6755  ...

$cooks   # Later
 [1]  0.11076  0.01459   ...
$dfits   # Later
 [1] -1.0508  0.3587  0.5523  ...

$correlation
```

```
  Not shown; correlation matrix of
       regression coefficients

$std.err
  Not shown; Std errors of coefficients

$cov.scaled
 Not shown; covariance matrix of regression
 coefficients (= S^2*$cov.unscaled)
$cov.unscaled
 Not shown; (X'X)^1
```

Test to see if all terms in ‘C’ can be dropped:

```
> n = nrow(x)
> p = ncol(x)+1
> SSE.full = d$std.dev^2*(n-p)
>
> # Reduced model, without columns 3,5,6,9 of X
> fit2 = lsfit(x[ ,-c(3,5,6,9)], conv)
> d2 = ls.diag(fit2)
> p2 = p-4
> SSE.red = d2$std.dev^2*(n-p2)
>
> F = ((SSE.red-SSE.full)/(p-p2))/(SSE.full/(n-p))
> p.to.drop.C = 1-pf(F, p-p2, n-p)
> cat("p-value of test is", p.to.drop.C, "\n")
p-value of test is 0.2138615
```

But we should have looked first at the validity of these fits.

```
> fits = cbind(1,x)%*%fit1$coef
> plot(fits,d$std.res, ylab="std.res")
```

[natheight=17.6279cm, natwidth=17.6718cm, height=8.4241 width=8.4438cm] C:/sw50/temp/graphics/acet1$_{f}ig2_{3.pdf}$

Hardly a 'normal' looking display. The high correlation between 'C' and 'T' should have been a warning!

```
> xtx = crossprod(cbind(1,x))
> det(xtx)
[1] 1.214396e+14
> solve(xtx)
Error in solve.default(xtx) : system is
computationally singular:
reciprocal condition number = 1.02379e-22
> diag(d$cov.unscaled)
Intercept         T         M         C       T*M
1.210e+07 2.930e+01 2.278e+01 1.343e+08 1.269e-05
T*C       M*C        T2        M2        C2
8.358e+01 1.051e+02 4.424e-06 1.680e-04 7.294e+07
```

- A regression model suffers from ‘multicollinearity’ if some of the regressors (or linear combinations of them) are highly correlated. This results in $\mathbf{X}'\mathbf{X}$ being difficult to invert (numerical instability), and in highly varied regression coefficients (so that they might change radically if a different sample at the same $\mathbf{x}$-values is taken).

- One indication of multicollinearity is if $|\mathbf{X}'\mathbf{X}|$ is very small. (This was certainly not the case with the acetylene data.) Another indicator is large values of the 'condition numbers', which are $\kappa_j = \lambda_1/\lambda_j$ when $\lambda_1 \geq \cdots \geq \lambda_p$ are the ordered eigenvalues of $\mathbf{X}'\mathbf{X}$. Values of $\kappa > 100$ (1000) indicate moderate (severe) numerical instability. For the acetylene data some $\kappa > 10^{20}$.

```
> v = eigen(xtx)$values; v
 [1] 3.543840e+13 6.843691e+08 1.133772e+05
 [4] 1.156081e+04 1.371599e+03 1.794139e+00
 [7] 1.046738e-02 4.513771e-04 3.203306e-05
 [10] 2.608609e-07
> max(v)/min(v)
[1] 1.358517e+20
```

5. Ridge regression; Weighted and Generalized Least Squares

- A first remedial measure is to standardize the variables: replace each column of $\mathbf{X}$ by that column minus its mean, divided by (($\sqrt{n-1}\times$ std.dev. of the column). The transformed $\mathbf{X}$ will now satisfy $\mathbf{X}'\mathbf{X} = \mathbf{R}$, where r_{ij} is the correlation between the i^{th} and j^{th} columns of the original $\mathbf{X}$. Transform Y in the same way and carry out the regression using these transformed variables. (The column of 1's has become a column of 0's, so eliminate the intercept, which is estimated by $\bar{y}$ and subtracted from each y_i.)

  – The resulting regression coefficients are called the 'standardized coefficients'.

  – Often this transformation is applied only to the linear terms, not the second or higher order terms. (Why is this sensible?)

- This transformation alone sometimes reduces the multicollinearity.

- Multicollinearity often results from the minimum eigenvalue $ch_{\text{min}}\left(\mathbf{X}'\mathbf{X}\right)$ being near zero. But the eigenvalues of $\mathbf{X}'\mathbf{X} + k\mathbf{I}_p$ are those of $\mathbf{X}'\mathbf{X}$ plus k; this indicates a way around the problem. The 'ridge' estimator starts in standardized form (so no intercept) and then replaces $\hat{\boldsymbol{\theta}} = \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}$ by
$$\hat{\boldsymbol{\theta}}_R(k) = \left(\mathbf{X}'\mathbf{X} + k\mathbf{I}_p\right)^{-1}\mathbf{X}'\mathbf{y}.$$
The number $k \geq 0$ is called a 'biasing constant' because it results in biased (but less highly varied) estimates.
    - $\hat{\boldsymbol{\theta}}_R(k)$ can also be defined as the solution to the problem of minimizing $\|\mathbf{y} - \mathbf{X}\boldsymbol{\theta}\|^2 + k\,\|\boldsymbol{\theta}\|^2$, in which large regression coefficients are penalized (assigned).

– It can be shown (Assignment 1) that the associated mean squared error, defined as

$$MSE(k) = E\left[\left\|\hat{\boldsymbol{\theta}}_R(k) - \boldsymbol{\theta}\right\|^2\right]$$
$$\left(= tr\left\{E\left[\left(\hat{\boldsymbol{\theta}}_R(k) - \boldsymbol{\theta}\right)\left(\hat{\boldsymbol{\theta}}_R(k) - \boldsymbol{\theta}\right)'\right]\right\}\right),$$

is given by

$$\begin{aligned} MSE(k) &= tr\left[\text{cov}\left(\hat{\boldsymbol{\theta}}_R(k)\right)\right] + \left\|\text{bias}\left[\hat{\boldsymbol{\theta}}_R(k)\right]\right\|^2 \\ &= \sigma_\varepsilon^2 tr\left[\left(\mathbf{X}'\mathbf{X} + k\mathbf{I}_p\right)^{-2}\mathbf{X}'\mathbf{X}\right] \\ &\quad + k^2\boldsymbol{\theta}'\left(\mathbf{X}'\mathbf{X} + k\mathbf{I}_p\right)^{-2}\boldsymbol{\theta}. \end{aligned}$$

As k increases, the variance component decreases and the bias component increases; there is a range of values of k for which $MSE(k) < MSE(0)$.

– A common (but controversial) way to choose an appropriate value of k is to examine the ‘ridge trace’ plots, which are plots of the components of $\hat{\boldsymbol{\theta}}_R(k)$ vs. k. These typically vary wildly for k near zero; one takes as k the value at which they begin to stabilize.

- See the ‘acetylene2’ script from the course website. The correlation transformation alone does not help matters much. But a ridge regression – see traces in plots (a), (b), (c) below – indicates that $k \approx .03$ is a suitable biasing constant. The R output and the traces suggest eliminating all terms involving ‘C’.

```
> # Do a regression with this value of k, using
’pseudovalues’:
> # Here the original data will be used
> k = .03
> xnew = cbind(1,x)
> px = ncol(xnew)
> fit = lsfit(rbind(xnew, sqrt(k)*diag(px)),
    c(conv, rep(0, px)), int=F)
> ls.print(fit)
Residual Standard Error=0.9428
R-Square=0.9994
F-statistic (df=10, 16)=2583.722
p-value=0
```

```
    Estimate Std.Err t-value Pr(>|t|)
      0.2264  5.4406  0.0416   0.9673
T    -0.1725  0.0260 -6.6359   0.0000
M     9.8492  2.0965  4.6978   0.0002
C     0.0892  5.4430  0.0164   0.9871
T*M  -0.0072  0.0016 -4.4386   0.0004
T*C  -0.0051  0.0482 -0.1054   0.9174
M*C  -0.3254  4.2212 -0.0771   0.9395
T2    0.0002  0.0000  8.3868   0.0000
M2   -0.0243  0.0089 -2.7370   0.0146
C2    0.0253  5.4434  0.0047   0.9963
```

- For the ridge regressions the standardized variable were used; (d) - (f) use the original variables. Eliminating 'C' and using $k = 0$ results in plots (e) and (f).

[natheight=7.1676in, natwidth=7.1676in, height=7.1952in, width=7.1952in] C:/sw50/temp/graphics/acet2$_{4.pdf}$

- Weighted and Generalized Least Squares. We might have evidence (obtained from residual plots, for instance) that the random errors ε are not uncorrelated, or not equally varied, and that instead cov$[\boldsymbol{\varepsilon}] = \sigma^2\boldsymbol{\Sigma} \neq \sigma_\varepsilon^2\mathbf{I}$. Suppose however that we can find a matrix, written $\boldsymbol{\Sigma}^{-1/2}$, such that

  $$\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\Sigma}\boldsymbol{\Sigma}^{-1/2'} = \mathbf{I}.$$

  Then the model $\mathbf{Y} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}$ can be transformed as

  $$\boldsymbol{\Sigma}^{-1/2}\mathbf{Y} = \boldsymbol{\Sigma}^{-1/2}\mathbf{X}\boldsymbol{\theta} + \boldsymbol{\Sigma}^{-1/2}\boldsymbol{\varepsilon}$$

  with cov$\left[\boldsymbol{\Sigma}^{-1/2}\boldsymbol{\varepsilon}\right] = \sigma^2\mathbf{I}$. Typically $\boldsymbol{\Sigma}^{-1/2}$ must be estimated; then the elements of $\hat{\boldsymbol{\Sigma}}^{-1/2}\mathbf{Y}$ become the new dependent variables and the rows of $\hat{\boldsymbol{\Sigma}}^{-1/2}\mathbf{X}$ become the new independent variables. This is called 'Generalized Least Squares', or 'Weighted Least Squares' if Σ is diagonal.

- Example: AR(1) data. Suppose the index 'i' represents time, and that the observations follow an AR(1) model common in economics:

$$\begin{aligned} Y_i &= \mathbf{x}_i'\boldsymbol{\theta} + \delta_i, \\ \delta_i &= \rho\delta_{i-1} + \varepsilon_i, \text{ with } |\rho| < 1. \quad (5.1) \end{aligned}$$

The ε_i in (5.1) are i.i.d. It is shown (in STAT 479: Time Series for instance) that then

$$\begin{aligned} \text{var}[Y_i] &= \text{var}[\delta_i] = \frac{\sigma_\varepsilon^2}{1-\rho^2}, \\ \text{corr}\left[Y_i, Y_j\right] &= \text{corr}\left[\delta_i, \delta_j\right] = \rho^{|i-j|}. \end{aligned}$$

An appropriate transformation is obvious from (5.1):

$$\begin{aligned} \hat{\boldsymbol{\Sigma}}^{-1/2}\mathbf{y} &= \begin{pmatrix} Y_1 \\ Y_2 - \hat{\rho}Y_1 \\ \vdots \\ Y_n - \hat{\rho}Y_{n-1} \end{pmatrix} \overset{def}{=} \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{pmatrix} = \mathbf{v}, \\ \hat{\boldsymbol{\Sigma}}^{-1/2}\mathbf{X} &= \begin{pmatrix} \mathbf{x}_1' \\ \mathbf{x}_2' - \hat{\rho}\mathbf{x}_1' \\ \vdots \\ \mathbf{x}_n' - \hat{\rho}\mathbf{x}_{n-1}' \end{pmatrix} \overset{def}{=} \begin{pmatrix} \mathbf{u}_1' \\ \mathbf{u}_2' \\ \vdots \\ \mathbf{u}_n' \end{pmatrix} = \mathbf{U}. \end{aligned}$$

(One might delete V_1 and $\mathbf{u}_1^{'}$.) Note that ρ is the correlation between successive observations Y_i, Y_{i-1}. The 'Cochrane-Orcutt' procedure is:

1. Fit an OLS model to the original data; obtain the residuals $\{e_i\}$.

2. Estimate ρ by the sample correlation of the pairs $\{(e_i, e_{i-1})\}_{i=2}^{n}$ (or the slope of a regression, through the origin, of $\{e_i\}_{i=2}^{n}$ on $\{e_{i-1}\}_{i=2}^{n}$).

3. Compute $\mathbf{v}$ and $\mathbf{U}$, estimate the parameters by $\hat{\boldsymbol{\theta}} = (\mathbf{U}'\mathbf{U})^{-1}\mathbf{U}'\mathbf{v}$; obtain new residuals $\{e_i\}$.

4. Repeat 2 and 3 if necessary; iterate until residual plots (against the index, or against the lag-1 values), or the 'Durbin-Watson test', indicate that the dependence has successfully been removed.

- When $\mathbf{\Sigma} = diag\left(\sigma_1^2, ..., \sigma_n^2\right)$, we have $\mathbf{\Sigma}^{-1/2} = diag\left(\sigma_1^{-1}, ..., \sigma_n^{-1}\right)$ and $\mathbf{\Sigma}^{-1}$ is called $\mathbf{W}$. The 'weights' w_i are the inverses of the (estimated) variances and the resulting regression of $\left\{\sqrt{w_i} y_i\right\}_{i=1}^n$ on $\left\{\sqrt{w_i} \mathbf{x}_i'\right\}_{i=1}^n$ results in the WLS estimate
$$\hat{\boldsymbol{\theta}}_{WLS} = \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1} \mathbf{X}'\mathbf{W}\mathbf{y}.$$

- Similarly, the GLS estimate can be written
$$\hat{\boldsymbol{\theta}}_{GLS} = \left(\mathbf{X}'\mathbf{\Sigma}^{-1}\mathbf{X}\right)^{-1} \mathbf{X}'\mathbf{\Sigma}^{-1}\mathbf{y}.$$

- **Delta method**: Let $\bar{X}$ be the average of i.i.d. observations X_i with mean μ and variance σ^2. Consider a function $f\left(\bar{X}\right)$. Expand around μ:
$$\begin{aligned} f\left(\bar{X}\right) &= f(\mu) + f'(\mu)\left(\bar{X} - \mu\right) \\ &\quad + \text{rem. of order } \left(\bar{X} - \mu\right)^2. \end{aligned}$$

Since $E\left[\left(\bar{X}-\mu\right)^2\right]=\sigma^2/n$, we can ignore this remainder for large n and conclude that

$$\begin{aligned} E\left[f\left(\bar{X}\right)\right] &\approx f\left(\mu\right), \\ \text{var}\left[f\left(\bar{X}\right)\right] &\approx E\left[\left\{f'\left(\mu\right)\left(\bar{X}-\mu\right)\right\}^2\right] \\ &= \left[f'\left(\mu\right)\right]^2\frac{\sigma^2}{n}. \end{aligned}$$

– One need not be working with ordinary averages – any estimate whose variance is of order $1/n$ will do. For instance if a residual plot of e against $\hat{Y}$ indicates that $\sigma_\varepsilon^2=\sigma_Y^2$ varies with μ_Y, an appropriate *variance stabilizing transformation* regresses $f\left(Y\right)$ on the X's, where f satisfies $f'\left(\mu\right)\sigma\left(\mu\right)=const.$ Example: if the spread in the plot indicates that $\sigma\left(\mu\right)\propto\mu$ (indicated by $s\left(e\right)\propto\hat{Y}$) then solve $f'\left(\mu\right)\mu=const.$ to get $f\left(Y\right)=\log Y$ (if $Y>0$). So regress $\log Y$ on the X's and check the residuals again.

6. Logistic regression through WLS or Maximum likelihood

- Logistic regression with repeat observations. Here the dependent variable Y_i is binary, representing the occurrence of some event (a 'cure', for instance):

  $$Y_i = \begin{cases} 1, & \text{if the event occurs on the } i^{th} \text{ trial,} \\ 0, & \text{otherwise.} \end{cases}$$

  Put $\pi_i = P(Y_i = 1)$. We want to investigate the manner in which π_i varies with covariates $\mathbf{x}_i$ at which Y_i is observed. But $Y_i \sim bin(1, \pi_i)$ is not normal, and the Y_i are not equally varied. A common approach is to make a *logistic transformation*. The logistic d.f. is $L(t) = 1/\left(1 + e^{-t}\right)$; it maps $(-\infty, \infty)$ into $(0, 1)$ and

  $$L^{-1}(\pi_i) = \log \frac{\pi_i}{1 - \pi_i}.$$

  We would like to model $L^{-1}(\pi_i)$ in terms of the covariates by taking as the data $L^{-1}(Y_i)$. This won't work (why not?) but we can do it if we have repeat observations $\left\{Y_{ij}\right\}_{j=1}^{n_i}$ at $\mathbf{x}_i$, $i = 1, ..., I$. First estimate π_i by $\hat{\pi}_i = \sum_j Y_{ij}/n_i$; then regress the 'logits'

$$v_i = \log \frac{\hat{\pi}_i}{1 - \hat{\pi}_i}$$

on the regressors in the model $v_i = \mathbf{x}_i' \boldsymbol{\theta} + \varepsilon_i$. Since v_i can be anything in $(-\infty, \infty)$ it might possibly look Normal, but are the ε_i equally varied?

– Apply the delta method with $\bar{X}$ replaced by $\hat{\pi}_i$, which is the average of n_i i.i.d. observations, each of which has mean π_i and variance $\pi_i (1 - \pi_i)$. The function $f(\hat{\pi}_i) = v_i$ has $f'(\hat{\pi}_i) = (\pi_i (1 - \pi_i))^{-1}$ and so

$$\text{var}[v_i] \approx \frac{1}{n_i \pi_i (1 - \pi_i)}.$$

The regression of v_i on $\mathbf{x}_i$ should then be done by WLS, with weights $w_i = n_i \hat{\pi}_i (1 - \hat{\pi}_i)$.

- Data setting: A market research company wishes to investigate the effectiveness of discount coupons. Coupons of varying values (5¢, 7¢,...,25¢) are given – to 500 people each – and the numbers of coupons redeemed after one month are recorded. How does the redemption rate depend on the coupon value?

```
# x = value of discount
# N = numbers of coupons redeemed

x = seq(from = 5, to = 25, by = 2)
N = c(100, 122, 147, 176, 211, 244,
        277, 310, 343, 372, 391)
p = N/500
logits = log(p/(1-p))

par(mfrow=c(1,2))

plot(x,logits)
fit = lm(logits ~x, weights = 500*p*(1-p))
lines(x, predict(fit))
```

```
plot(predict(fit), ls.diag(fit)$stud.res,
xlab = "fits", ylab = "stud. res.")
```

[natheight=7.1676in, natwidth=7.1676in, height=5.7614in, width=5.7614in]
C:/sw50/temp/graphics/logistic$_{5.pdf}$
Data, fitted regression line, and studentized residuals in logistic regression example.

- Logistic regression without replicates – use maximum likelihood. Our model is that one observes independent r.v.s $Y_1, ..., Y_n$ with $Y_i \sim bin(1, \pi_i)$ and $\log \frac{\pi_i}{1-\pi_i} = \mathbf{x}_i' \boldsymbol{\theta}$, implying

$$\pi_i = \frac{e^{\mathbf{x}_i' \boldsymbol{\theta}}}{1 + e^{\mathbf{x}_i' \boldsymbol{\theta}}};\ 1 - \pi_i = \frac{1}{1 + e^{\mathbf{x}_i' \boldsymbol{\theta}}}.$$

With data $(y_1, ..., y_n)$ the log-likelihood is

$$\begin{aligned} l(\boldsymbol{\theta}) &= \log\left\{\prod_{i=1}^{n} \pi_i^{y_i}(1-\pi_i)^{1-y_i}\right\} \\ &= \sum_{i=1}^{n} y_i \log \pi_i + (1-y_i)\log(1-\pi_i) \\ &= \sum_{i=1}^{n} y_i \log\frac{\pi_i}{1-\pi_i} + \sum_{i=1}^{n} \log(1-\pi_i) \\ &= \sum_{i=1}^{n} y_i \mathbf{x}_i'\boldsymbol{\theta} - \sum_{i=1}^{n} \log\left(1+e^{\mathbf{x}_i'\boldsymbol{\theta}}\right). \end{aligned}$$

The likelihood equations are $\dot{l}(\boldsymbol{\theta})$ $(= \left(\frac{\partial l}{\partial \boldsymbol{\theta}}\right)') = \mathbf{0}$:

$$\begin{aligned} \frac{\partial l}{\partial \boldsymbol{\theta}} &= \sum_{i=1}^{n} y_i \mathbf{x}_i' - \sum_{i=1}^{n} \frac{e^{\mathbf{x}_i'\boldsymbol{\theta}}}{1+e^{\mathbf{x}_i'\boldsymbol{\theta}}}\mathbf{x}_i' \\ &= \sum_{i=1}^{n} (y_i - \pi_i)\,\mathbf{x}_i' = (\mathbf{y} - \boldsymbol{\pi}(\boldsymbol{\theta}))'\,\mathbf{X}. \end{aligned}$$

The are generally solved by Newton-Raphson. We are to find a starting value $\boldsymbol{\theta}_{(0)}$ and iterate to convergence:

$$\boldsymbol{\theta}_{(k+1)} = \boldsymbol{\theta}_{(k)} - \left[\ddot{l}\left(\boldsymbol{\theta}_{(k)}\right)\right]^{-1} \dot{l}\left(\boldsymbol{\theta}_{(k)}\right).$$

A calculation gives

$$\ddot{l}\left(\boldsymbol{\theta}_{(k)}\right) = -\mathbf{X}'\mathbf{W}_{(k)}\mathbf{X},$$

where $\mathbf{W}_{(k)} = diag\left(\cdots, \pi_i\left(\boldsymbol{\theta}_{(k)}\right)\left(1 - \pi_i\left(\boldsymbol{\theta}_{(k)}\right)\right), \cdots\right)$.

This results in

$$\begin{aligned}\boldsymbol{\theta}_{(k+1)} &= \boldsymbol{\theta}_{(k)} + \left(\mathbf{X}'\mathbf{W}_{(k)}\mathbf{X}\right)^{-1}\mathbf{X}'\left(\mathbf{y}-\boldsymbol{\pi}\left(\boldsymbol{\theta}_{(k)}\right)\right) \\ &= \left(\mathbf{X}'\mathbf{W}_{(k)}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}_{(k)}\mathbf{z}_{(k)}\end{aligned}$$

where

$$\begin{aligned}\mathbf{z}_{(k)} &= \mathbf{W}_{(k)}^{-1}\left(\mathbf{y}-\boldsymbol{\pi}\left(\boldsymbol{\theta}_{(k)}\right)\right)+\mathbf{X}\boldsymbol{\theta}_{(k)} \\ &= \begin{pmatrix} \vdots \\ \frac{y_i-\pi_i\left(\boldsymbol{\theta}_{(k)}\right)}{\pi_i\left(\boldsymbol{\theta}_{(k)}\right)\left(1-\pi_i\left(\boldsymbol{\theta}_{(k)}\right)\right)}+\mathbf{x}_i'\boldsymbol{\theta}_{(k)} \\ \vdots \end{pmatrix}.\end{aligned}$$

Thus the algorithm is to repeatedly do WLS regressions of $\mathbf{z}_{(k)}$ on $\mathbf{X}$, until convergence:

(i) Calculate $\boldsymbol{\theta}_{(0)}$.

(ii) For $k = 0, 1, \ldots$calculate (in R notation)

$$\begin{aligned}\boldsymbol{\pi}_{(k)} &= \text{exp}\left(\mathbf{X}\boldsymbol{\theta}_{(k)}\right)/\left(1+\text{exp}\left(\mathbf{X}\boldsymbol{\theta}_{(k)}\right)\right), \\ \mathbf{w}_{(k)} &= \boldsymbol{\pi}_{(k)} * \left(1-\boldsymbol{\pi}_{(k)}\right), \\ \mathbf{z}_{(k)} &= \left[\left(\mathbf{y}-\boldsymbol{\pi}_{(k)}\right)/\mathbf{w}_{(k)}\right]+\mathbf{X}\boldsymbol{\theta}_{(k)}, \\ \boldsymbol{\theta}_{(k+1)} &= \texttt{lsfit}(\mathbf{X},\mathbf{z}_{(k)},\texttt{weights}=\mathbf{w}_{(k)},\texttt{int}=\texttt{F})\$\texttt{coef}.\end{aligned}$$

7. Influence measures

- Effect of outliers and highly influential values

[natheight=18.2056cm, natwidth=18.2056cm, height=11.1764cm, width=12.1825cm]
C:/sw50/temp/graphics/influence$_{fig}1_{6.pdf}$
Simulated data: $Y = x + \varepsilon$ with, on the left, one additional observation which is both highly influential (extreme $x-$ value) and has an outlying $y-$ value. Plot on right is after removal of this point.

[natheight=18.2056cm, natwidth=18.2056cm, height=11.4444cm, width=12.4505cm]
C:/sw50/temp/graphics/influence$_{fig}2_{7.pdf}$
Residuals from fits to full data set. The "bad" point does not show up as an unusually large residual in the LS fit, but the "good" points show an alarming trend. In higher dimensions especially one cannot count on the residuals to reveal problems with outliers or high leverages.

- We can compute measures of influence which help to identify ‘bad’ points; we can also use more robust estimation methods which are less highly influenced by such points. Right now we’ll look at measures of influence.

- Hat matrix diagnostics. The residuals are

  $$\mathbf{e} = \mathbf{y} - \hat{\mathbf{y}} = (\mathbf{I} - \mathbf{H})\,\mathbf{y}$$

  hence

  $$\begin{aligned} \text{cov}\,[\mathbf{e}] &= \sigma_\varepsilon^2\,(\mathbf{I} - \mathbf{H})\,,\ \text{cov}\,[\hat{\mathbf{y}}] = \sigma_\varepsilon^2\mathbf{H}; \\ \text{var}\,[e_i] &= \sigma_\varepsilon^2\,(1 - h_{ii})\,, \text{var}\,[\hat{y}_i] = \sigma_\varepsilon^2 h_{ii}. \end{aligned}$$

  Thus $0 \leq h_{ii} \leq 1$ (with $\sum h_{ii} = tr\,(\mathbf{H}) = p$). A value of h_{ii} near 1 results (with high probability) in a very small value of e_i; thus $\hat{y}_i$ is forced to be nearly equal to y_i. For this reason the h_{ii} are called *leverages*. On R they are printed out as the ‘hat’ component of `ls.diag`, a ‘rule of thumb’ is that a value $h_{ii} > 2\bar{h} = 2p/n$ is a cause for alarm.

- Deleted and studentized residuals. The i^{th} *deleted residual* is

$$d_i = y_i - \hat{y}_{i(i)},$$

where $\hat{y}_{i(i)}$ is the predicted value of Y_i, computed from the sample with $(\mathbf{x}_i, y_i)$ removed (so that the prediction is not influenced by the i^{th} case. The *studentized residual* is

$$d_i^* = \frac{d_i}{s\left(d_i\right)} = \cdots = \frac{e_i}{S_{(i)}\sqrt{1-h_{ii}}},$$

where $S_{(i)}$ is computed from the reduced sample and '$\cdots$' refers to some algebra to be outlined later. In a similar manner,

$$\begin{aligned}(n-p-1)\, S_{(i)}^2 &\overset{def}{=} \sum_{j\neq i}\left(y_j - \hat{y}_{j(i)}\right)^2 \\ &= (n-p)\, S^2 - \frac{e_i^2}{1-h_{ii}}.\end{aligned}$$

Thus d_i^* can be computed <u>without</u> doing all n reduced regressions. Note that $d_i^* \sim t_{n-p-1}$, and so $\left|d_i^*\right|$ is typically compared to $t_{n-p-1}\left(.95\right)$, larger values indicating an outlying

y-value. In residual plots one generally plots the studentized residuals (rather than the *standardized residuals* $r_i = \frac{e_i}{S\sqrt{1-h_{ii}}}$, which are *not* t_{n-p} – why not?).

- Deleted fits. The i^{th} *deleted fitted value* is
$$dfit_i = \frac{\hat{y}_i - \hat{y}_{i(i)}}{S_{(i)}\sqrt{h_{ii}}} = \cdots = d_i^* \left(\frac{h_{ii}}{1-h_{ii}} \right)^{1/2}.$$
From the final expression we see that this is large if d_i^* is large (an outlying y-value) or if h_{ii} is large (an influential $\mathbf{x}$-value). Its absolute value is typically compared to min $\left(1, 2\sqrt{\frac{p}{n}}\right)$.

- Cook's statistic.
$$D_i = \frac{\left(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_{(i)}\right)' \mathbf{X}'\mathbf{X} \left(\hat{\boldsymbol{\theta}} - \hat{\boldsymbol{\theta}}_{(i)}\right)}{pS^2} = \cdots = r_i^2 \frac{h_{ii}}{p\,(1-h_{ii})}.$$
This is compared to $F^p_{n-p}(.5)$, a larger value indicating that when the i^{th} case is deleted, the vector of regression coefficients moves out of a 50% confidence ellipsoid computed from $\hat{\boldsymbol{\theta}}$.

- These and many other such measures are discussed in 'Influential Observations, High Leverage Points, and Outliers in Linear Regression' by Chatterjee & Hadi; available on course website.

- See the 'acetylene' output from Lecture 4 – all of these are components of `ls.diag`.

<u>'Some algebra' to be outlined now</u>:

- Useful identity:
$$\left(\mathbf{I}-\mathbf{ab}'\right)^{-1}=\mathbf{I}+\frac{\mathbf{ab}'}{1-\mathbf{b}'\mathbf{a}}.$$
Proof: $\mathbf{I}-\mathbf{ab}'\times$ RHS $=\ldots=\mathbf{I}$.
Motivation: ...

- Similarly, $\left|\mathbf{I}-\mathbf{ab}'\right|=\left|\mathbf{I}-\mathbf{b}'\mathbf{a}\right|=1-\mathbf{b}'\mathbf{a}$.

- More generally, $\left(\mathbf{I}-\mathbf{AB}\right)^{-1}=\mathbf{I}+\mathbf{A}\left(\mathbf{I}-\mathbf{BA}\right)^{-1}\mathbf{B}$ and $\left|\mathbf{I}-\mathbf{AB}\right|=\left|\mathbf{I}-\mathbf{BA}\right|$.

- Suppose we delete one row – the first, say – from the $\mathbf{X}$-matrix. How does this affect $\left(\mathbf{X}'\mathbf{X}\right)^{-1}$? Write

$$\begin{aligned}
\mathbf{X} &= \binom{\mathbf{x}_1'}{\mathbf{X}_{(1)}}, \\
\mathbf{X}'\mathbf{X} &= \mathbf{X}_{(1)}'\mathbf{X}_{(1)} + \mathbf{x}_1\mathbf{x}_1',
\end{aligned}$$

so that

$$\begin{aligned}
&\left[\mathbf{X}_{(1)}'\mathbf{X}_{(1)}\right]^{-1} \\
&= \left[\mathbf{X}'\mathbf{X} - \mathbf{x}_1\mathbf{x}_1'\right]^{-1} \\
&= \left[\mathbf{X}'\mathbf{X}\left\{\mathbf{I} - \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_1\mathbf{x}_1'\right\}\right]^{-1} \\
&= \left\{\mathbf{I} + \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_1\mathbf{x}_1'}{1 - \mathbf{x}_1'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_1}\right\}\left(\mathbf{X}'\mathbf{X}\right)^{-1} \\
&= \left(\mathbf{X}'\mathbf{X}\right)^{-1} + \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_1\mathbf{x}_1'\left(\mathbf{X}'\mathbf{X}\right)^{-1}}{1 - h_{11}}.
\end{aligned}$$

Clearly $\mathbf{X}_{(1)}'\mathbf{y}_{(1)} = \mathbf{X}'\mathbf{y} - \mathbf{x}_1 y_1$, so that the regression coefficients computed from the reduced sample are

$$\begin{aligned}
\hat{\theta}_{(1)} &= \left[\mathbf{X}_{(1)}^{\prime}\mathbf{X}_{(1)}\right]^{-1}\mathbf{X}_{(1)}^{\prime}\mathbf{y}_{(1)} \\
&= \ldots \\
&= \hat{\theta} - \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_1 e_1}{1-h_{11}};
\end{aligned}$$

and in general

$$\hat{\theta}_{(i)} = \hat{\theta} - \frac{\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{x}_i e_i}{1-h_{ii}}.$$

- Apply to Cook's statistic:

$$\begin{aligned}
D_i &= \frac{\left(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_{(i)}\right)'\mathbf{X}'\mathbf{X}\left(\hat{\boldsymbol{\theta}}-\hat{\boldsymbol{\theta}}_{(i)}\right)}{pS^2} \\
&= \frac{\left(\frac{\mathbf{x}_i e_i}{1-h_{ii}}\right)'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\left(\frac{\mathbf{x}_i e_i}{1-h_{ii}}\right)}{pS^2} \\
&= \frac{e_i^2}{\left(1-h_{ii}\right)^2 pS^2}h_{ii} \\
&= \left(\frac{e_i}{S\sqrt{1-h_{ii}}}\right)^2\frac{h_{ii}}{p\left(1-h_{ii}\right)}.
\end{aligned}$$

- Studentized residuals:

  $$\hat{y}_{i(i)} = \mathbf{x}_i'\hat{\boldsymbol{\theta}}_{(i)} = ... = \hat{y}_i - \frac{h_{ii}e_i}{1-h_{ii}},$$

  so

  $$d_i = y_i - \hat{y}_{i(i)} = \frac{e_i}{1-h_{ii}},$$

  with standard deviation

  $$\sigma\left(d_i\right) = \frac{\sigma\left(e_i\right)}{1-h_{ii}} = \frac{\sigma_\varepsilon}{\sqrt{1-h_{ii}}};$$

  thus (using $S_{(i)}$ to estimate σ_ε)

  $$d_i^* = \frac{d_i}{s\left(d_i\right)} = \frac{e_i}{S_{(i)}\sqrt{1-h_{ii}}}.$$

- Deleted fits: From the preceding,

  $$dfit_i = \frac{\hat{y}_i - \hat{y}_{i(i)}}{S_{(i)}\sqrt{h_{ii}}} = \frac{\frac{h_{ii}e_i}{1-h_{ii}}}{S_{(i)}\sqrt{h_{ii}}} = ... = d_i^*\left(\frac{h_{ii}}{1-h_{ii}}\right)^{1/2}.$$

- That $\left(n-p-1\right)S_{(i)}^2 = \left(n-p\right)S^2 - \frac{e_i^2}{1-h_{ii}}$ is left as an exercise (Assignment 1).

Part II

Nonlinear Regression

8. Nonlinear models; Gauss-Newton algorithm

- Good reading material for these lectures on nonlinear regression:

    1. Bates & Watts Chapters 2, 3.1 - 3.6, 6.1 (in the first edition).

    2. Seber & Wild Chapter 5.1 - 5.4.

- Recall from Lecture 1: In pharmacology and elsewhere the output (Y) of a chemical reaction may depend on the input x, random error ε and parameters θ_1, θ_2 according to a 'Michaelis-Menten' model
$$Y = \frac{\theta_1 x}{\theta_2 + x} + \varepsilon.$$
Note horizontal asymptote at θ_1, 'halfway point' is $x = \theta_2$. Symbolically,
$$Y_i = f\left(\boldsymbol{\theta}; x_i\right) + \varepsilon_i,\ i = 1, ..., n.$$

The function $f(\boldsymbol{\theta}; x) = \frac{\theta_1 x}{\theta_2 + x}$ is a non-linear function of $\boldsymbol{\theta}$. Formally, this means that

$$\dot{\mathbf{f}}'(\boldsymbol{\theta}; x_i) = \left(\frac{\partial f(\boldsymbol{\theta}; x_i)}{\partial \theta_1}, \frac{\partial f(\boldsymbol{\theta}; x_i)}{\partial \theta_2}\right)$$

depends on $\boldsymbol{\theta}$. Hence this is a *non-linear regression model*. With

$$\mathbf{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix}, \boldsymbol{\eta}(\boldsymbol{\theta}) = \begin{pmatrix} f(\boldsymbol{\theta}; x_1) \\ \vdots \\ f(\boldsymbol{\theta}; x_n) \end{pmatrix}, \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

we have

$$\mathbf{Y} = \boldsymbol{\eta}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}.$$

- Example (‘Rumford1’ in Bates & Watts): The expected temperature of an object, allowed to cool from an initial temperature of 130° to an ambient temperature of 60° is, after time t, given by $E[Y] = 60 + 70e^{-\theta t}$.

- The MM model is 'transformably linear': Ignore the errors, and write
$$\begin{aligned} \frac{1}{f} &= \beta_0 + \beta_1 u \text{ for} \\ \beta_0 &= \frac{1}{\theta_1}, \beta_1 = \frac{\theta_2}{\theta_1}, u = \frac{1}{x}. \end{aligned}$$
So one can regress $1/y$ on u to get initial estimates. But of course
$$\frac{1}{y} \neq \beta_0 + \beta_1 u + \text{ error.}$$

- The MM model is also 'conditionally linear' in θ_1, given θ_2: if we are given θ_2 we can put
$$z = \frac{x}{\theta_2 + x}$$
and the model becomes $Y = \theta_1 z + \varepsilon$ – SLR through the origin. Raises the possibility of a mixture of estimation techniques ...

- Gauss-Newton method. We aim to minimize
$$S(\boldsymbol{\theta}) = \|\mathbf{y} - \boldsymbol{\eta}(\boldsymbol{\theta})\|^2 .$$

Notation:

$$\dot{\boldsymbol{\eta}}\left(\boldsymbol{\theta}\right)=\begin{pmatrix}\dot{\mathbf{f}}'\left(\boldsymbol{\theta};\mathbf{x}_1\right)\\ \vdots\\ \dot{\mathbf{f}}'\left(\boldsymbol{\theta};\mathbf{x}_n\right)\end{pmatrix}\ \text{(so } \left[\dot{\boldsymbol{\eta}}\left(\boldsymbol{\theta}\right)\right]_{ij}=\frac{\partial f\left(\boldsymbol{\theta};\mathbf{x}_i\right)}{\partial\theta_j}\text{).}$$

Let $\boldsymbol{\theta}_{(0)}$ be an initial estimate. Recall Taylor's Theorem, by which each component $\eta_i\left(\boldsymbol{\theta}\right)=f\left(\boldsymbol{\theta};\mathbf{x}_i\right)$ of $\boldsymbol{\eta}\left(\boldsymbol{\theta}\right)$ can be expanded as

$$\begin{aligned}\eta_i\left(\boldsymbol{\theta}\right) &= f\left(\boldsymbol{\theta}_{(0)};\mathbf{x}_i\right)+\dot{\mathbf{f}}'\left(\boldsymbol{\theta}_{(0)};\mathbf{x}_i\right)\left(\boldsymbol{\theta}-\boldsymbol{\theta}_{(0)}\right)\\ &\quad+\frac{1}{2}\left(\boldsymbol{\theta}-\boldsymbol{\theta}_{(0)}\right)'\ddot{\mathbf{f}}\left(\tilde{\boldsymbol{\theta}};\mathbf{x}_i\right)\left(\boldsymbol{\theta}-\boldsymbol{\theta}_{(0)}\right),\end{aligned}$$

for some $\tilde{\boldsymbol{\theta}}$ between $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_{(0)}$. We apply this and ignore the Hessian, obtaining the linear approximation

$$\begin{aligned}\boldsymbol{\eta}\left(\boldsymbol{\theta}\right) &\approx \boldsymbol{\eta}\left(\boldsymbol{\theta}_{(0)}\right)+\dot{\boldsymbol{\eta}}\left(\boldsymbol{\theta}_{(0)}\right)\left(\boldsymbol{\theta}-\boldsymbol{\theta}_{(0)}\right)\\ &= \boldsymbol{\eta}_{(0)}+\mathbf{V}_{(0)}\boldsymbol{\delta}\text{, say.}\end{aligned}$$

Then

$$\mathbf{Y}\approx\boldsymbol{\eta}_{(0)}+\mathbf{V}_{(0)}\boldsymbol{\delta}+\boldsymbol{\varepsilon}, \tag{8.1}$$

leading to

$$\hat{\boldsymbol{\delta}}=\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}_{(0)}=\left(\mathbf{V}_{(0)}'\mathbf{V}_{(0)}\right)^{-1}\mathbf{V}_{(0)}'\left(\mathbf{y}-\boldsymbol{\eta}_{(0)}\right).$$

This $\hat{\boldsymbol{\theta}}$ becomes the next iterate:

$$\boldsymbol{\theta}_{(1)} = \boldsymbol{\theta}_{(0)} + \left(\mathbf{V}_{(0)}'\mathbf{V}_{(0)}\right)^{-1}\mathbf{V}_{(0)}'\left(\mathbf{y} - \boldsymbol{\eta}_{(0)}\right),$$

and in general

$$\boldsymbol{\theta}_{(k+1)} = \boldsymbol{\theta}_{(k)} + \left(\mathbf{V}_{(k)}'\mathbf{V}_{(k)}\right)^{-1}\mathbf{V}_{(k)}'\left(\mathbf{y} - \boldsymbol{\eta}_{(k)}\right).$$

Note then that the residuals $\mathbf{y}-\boldsymbol{\eta}_{(k)}$ from the k^{th} stage are being regressed on the columns of $\mathbf{V}_{(k)}$; the resulting regression coefficients are added to $\boldsymbol{\theta}_{(k)}$ to get $\boldsymbol{\theta}_{(k+1)}$.

- Now check that $S\left(\boldsymbol{\theta}_{(k+1)}\right) < S\left(\boldsymbol{\theta}_{(k)}\right)$; if not replace $\boldsymbol{\theta}_{(k+1)}$ by $\boldsymbol{\theta}_{(k)} + \lambda\left(\boldsymbol{\theta}_{(k+1)} - \boldsymbol{\theta}_{(k)}\right)$ for $\lambda = 1/2, 1/4, ...$ until there is a decrease in $S(\cdot)$.

- This is what is done in R (with of course the QR-decompostion of $\mathbf{V}_{(k)}$) in the function `nls(...)`.

- When to stop? After discussing several other possibilities – relative change $\left\|\boldsymbol{\theta}_{(k+1)}-\boldsymbol{\theta}_{(k)}\right\| / \left\|\boldsymbol{\theta}_{(k)}\right\|$ in the coefficients, or relative change in S, Bates & Watts suggest (and this is used in R) the orthogonality of the residual vector $\mathbf{y}-\boldsymbol{\eta}(\boldsymbol{\theta})$ to the (tangent to the) expectation surface $\boldsymbol{\eta}(\boldsymbol{\theta})$, when evaluated at $\hat{\boldsymbol{\theta}}$. Motivation: at a critical point $\hat{\boldsymbol{\theta}}$ of $S(\boldsymbol{\theta})$, one has
$$\left(\mathbf{y}-\boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right)\right)'\dot{\boldsymbol{\eta}}\left(\hat{\boldsymbol{\theta}}\right)=\mathbf{0}'. \tag{8.2}$$
To assess this, consider the linear approximation as at (8.1): $\mathbf{z}=\mathbf{V}\boldsymbol{\delta}+\boldsymbol{\varepsilon}$, with $\mathbf{z}=\mathbf{y}-\boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right)$ and $\mathbf{V}=\mathbf{V}\left(\hat{\boldsymbol{\theta}}\right)=\dot{\boldsymbol{\eta}}\left(\hat{\boldsymbol{\theta}}\right)$. The LR test of the hypothesis that $\boldsymbol{\delta}=\mathbf{0}$ is based on
$$F=\left(\frac{SSE_{Red}-SSE_{Full}}{p}\right) / \left(\frac{SSE_{Full}}{n-p}\right).$$
The numerator of the F is
$$\mathbf{z}'\mathbf{z}-\mathbf{z}'\left(\mathbf{I}-\mathbf{V}\left(\mathbf{V}'\mathbf{V}\right)^{-1}\mathbf{V}'\right)\mathbf{z}=\mathbf{z}'\left(\mathbf{V}\left(\mathbf{V}'\mathbf{V}\right)^{-1}\mathbf{V}'\right)\mathbf{z},$$
which at a critical point will $=$ 0 since $\mathbf{z}'\mathbf{V}=\left(\mathbf{y}-\boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right)\right)'\dot{\boldsymbol{\eta}}\left(\hat{\boldsymbol{\theta}}\right)$.

- The stopping rule is thus: Compute $\boldsymbol{\theta}_{(k)}$, $\mathbf{V}_{(k)}$ and $\mathbf{z} = \mathbf{y} - \boldsymbol{\eta}\left(\boldsymbol{\theta}_{(k)}\right)$. Fit the model $\mathbf{z} = \mathbf{V}_{(k)}\boldsymbol{\delta} + \boldsymbol{\varepsilon}$. Use the output to run the F-test of *H*: $\boldsymbol{\delta} = \mathbf{0}$. If $\sqrt{F_{obs}} < .001$, stop. If not, put $\boldsymbol{\theta}_{(k+1)} = \boldsymbol{\theta}_{(k)} + \hat{\boldsymbol{\delta}}$ and repeat.

- 'Linear approximation' inferences are made by expanding $\mathbf{y} - \boldsymbol{\eta}(\boldsymbol{\theta})$ as

  $$\boldsymbol{\varepsilon} = \mathbf{y} - \boldsymbol{\eta}(\boldsymbol{\theta}) \approx \mathbf{y} - \boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right) - \dot{\boldsymbol{\eta}}\left(\hat{\boldsymbol{\theta}}\right)\left(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}\right);$$

  $$\text{i.e. } \mathbf{y} - \boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right) \approx \boldsymbol{\varepsilon} - \hat{\mathbf{V}}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right),$$

  where $\hat{\mathbf{V}} = \mathbf{V}\left(\hat{\boldsymbol{\theta}}\right)$. Then (8.2) has the 'solution'

  $$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right)^{-1}\hat{\mathbf{V}}'\boldsymbol{\varepsilon}$$

  and the approximation is

  $$\hat{\boldsymbol{\theta}} \overset{d}{\approx} N\left(\boldsymbol{\theta}, \sigma_{\varepsilon}^{2}\left(\mathbf{V}'\mathbf{V}\right)^{-1}\right),$$

  with $\mathbf{V} = \mathbf{V}(\boldsymbol{\theta})$ estimated by $\hat{\mathbf{V}}$. From this, single parameter inferences can be made exactly as in Lecture 3, with $\mathbf{X}$ replaced by $\hat{\mathbf{V}}$. Ellipsoidal confidence regions on $\boldsymbol{\theta}$ can also be obtained as there.

- Linear approximation inferences on $f\left(\boldsymbol{\theta};\mathbf{x}_0\right) = E\left[Y|\mathbf{x}_0\right]$: First expand

  $$\begin{aligned} f\left(\boldsymbol{\theta};\mathbf{x}_0\right) &\approx f\left(\hat{\boldsymbol{\theta}};\mathbf{x}_0\right) + \dot{f}'\left(\hat{\boldsymbol{\theta}};\mathbf{x}_0\right)\left(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}\right) \\ &= f\left(\hat{\boldsymbol{\theta}};\mathbf{x}_0\right) + \mathbf{v}_0^{'}\left(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}\right). \end{aligned}$$

  Then

  $$f\left(\hat{\boldsymbol{\theta}};\mathbf{x}_0\right) \overset{d}{\approx} N\left(f\left(\boldsymbol{\theta};\mathbf{x}_0\right), \sigma_\varepsilon^2 \mathbf{v}_0^{'}\left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right)^{-1}\mathbf{v}_0\right),$$

  leading to a confidence interval

  $$f\left(\hat{\boldsymbol{\theta}};\mathbf{x}_0\right) \pm t_{n-p}\left(1-\frac{\alpha}{2}\right) S\sqrt{\mathbf{v}_0^{'}\left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right)^{-1}\mathbf{v}_0}$$

  where $S^2 = \left\|\mathbf{y}-\boldsymbol{\eta}\left(\hat{\boldsymbol{\theta}}\right)\right\|^2/\left(n-p\right)$. For simultaneous confidence intervals on *all* $f\left(\boldsymbol{\theta};\mathbf{x}_0\right)$, replace t_{n-p} by $\sqrt{pF_{n-p}^p}$.

- These linear approximation inferences can be very misleading; likelihood based inferences are generally recommended instead.

- R output for fit of a Michaelis-Menten model to the ‘puromycin’ data. Response Y is the ‘velocity’ of a reaction starting with a concentration x of ‘treated’ radioactive material.

```
> # Fit Michaelis-Menten data from Bates & Watts
  ... put the data (conc and vel) into
   a dataframe 'Micmen'...
> lin.params = lsfit(x=1/Micmen$conc,
       y=1/Micmen$vel)$coef
> theta1.start = 1/lin.params[1];
  theta2.start = lin.params[2]/lin.params[1]
> starting.values = list(theta1.start,theta2.start)

Starting values are
theta1.start = 195.8027
theta2.start = 0.04840653

> fit1 = nls(vel~theta1*conc/(theta2+conc),Micmen,
 start = starting.values, trace=T)
```

```
Tracing nls(vel ~theta1 * conc/(theta2 + conc),
 Micmen, start = starting.values,  .... on entry
1920.643 :  195.80270885   0.04840653
1207.887 :  210.888516     0.061361
1195.604 :  212.49074892   0.06380847
1195.450 :  212.66411803   0.06409054
1195.449 :  212.68183920   0.06411831
1195.449 :  212.683560     0.064121

> print(fit1)
Nonlinear regression model
  model:  vel ~theta1 * conc/(theta2 + conc)
   data:  Micmen
    theta1     theta2
212.683560   0.064121
 residual sum-of-squares: 1195.449
```

```
> print(summary(fit1))

Formula: vel ~theta1 * conc/(theta2 + conc)

Parameters:
        Estimate Std. Error t value Pr(>|t|)
theta1 2.127e+02  6.947e+00  30.615 3.24e-11 ***
theta2 6.412e-02  8.281e-03   7.743 1.57e-05 ***
---
Residual standard error: 10.93 on
   10 degrees of freedom

> # Define the response function AND its gradient:
> velocity = function(conc,theta1,theta2)  {
 velocity = theta1*conc/(theta2+conc)
 dvel.theta1 = conc/(theta2+conc)
 dvel.theta2 = -theta1*conc/((theta2+conc)^2)
 attr(velocity, "gradient") =
    cbind(dvel.theta1, dvel.theta2)
 velocity}
> fit2 = nls(vel~velocity(conc,theta1,theta2),
  Micmen,start=starting.values, trace=T)
```

```
> print(fit2)
Nonlinear regression model
  model:  vel ~velocity(conc, theta1, theta2)
  data:  Micmen
    theta1     theta2
212.683560   0.064121
 residual sum-of-squares: 1195.449

print(summary(fit2))

Formula: vel ~velocity(conc, theta1, theta2)

Parameters:
        Estimate Std. Error t value Pr(>|t|)
theta1 2.127e+02  6.947e+00  30.615 3.24e-11 ***
theta2 6.412e-02  8.281e-03   7.743 1.57e-05 ***
---
Residual standard error: 10.93 on
    10 degrees of freedom
```

```
> #Get the derivative matrix, as a function and
> # evaluated at the final estimates:
>
> V = function(theta1,theta2) {
  attr(velocity(conc,theta1,theta2),"gradient")
  }
> coefs = coef(fit2)
> resids = residuals(fit2)
> V.hat = V(coefs[1], coefs[2])

> print(V.hat)
      dvel.theta1 dvel.theta2
 [1,]   0.2377528   -601.1116
            ...         ...
[12,]   0.9449190   -172.6356
```

A linear regression of residuals(fit2) on the columns of the derivative matrix V.hat gives the following output:

```
> fin.lin.fit = lsfit(x=V.hat,y=resids,intercept=F)
> ls.print(fin.lin.fit)
```

```
Residual Standard Error=10.9337
R-Square=0
F-statistic (df=2, 10)=0
 ....
> #Prepare plot of estimated vel vs.\ conc on a
 grid of values, with data points on the same
 axes; include simultaneous confidence bands.

> conc.grid = seq(from=0,to=1.2,by=.02)
> vel.plot = velocity(conc.grid,coefs[1],coefs[2])
> vecnorm = function(vec) sqrt(crossprod(vec))
>
# Get half width of confidence band:
half.width = function(x, alpha)  {
 quant = qf(1-alpha, 2, 10)
 sigmahat = vecnorm(resids)/sqrt(10)
 R1 = qr.R(fin.lin.fit$qr)
 v0 = attr(velocity(x,coefs[1],coefs[2]),"gradient"
 norm = vecnorm(solve(t(R1),t(v0)))
 half.width = sigmahat*norm*sqrt(2*quant)
 half.width
  }
```

```
>
> alpha = .05
> half = NULL; for (x in conc.grid)
   half = c(half, half.width(x, alpha))
> upper.limits = vel.plot + half
> lower.limits = vel.plot - half
>
> par(oma=c(8,0,0,0))
> plot(x=conc.grid, y=vel.plot, type = "l",
 ylim = c(0,250), xlab = "concentration",
 ylab = "velocity")
> points(conc,vel,pch="*")
> lines(conc.grid, upper.limits, lty=2)
> lines(conc.grid, lower.limits, lty=2)
> mtext("...", side=1, outer=T)
```

[natheight=18.2056cm, natwidth=18.2056cm, height=11.013 width=14.6339cm] C:/sw50/temp/graphics/puromycin$_{8.pdf}$

9. Likelihood regions

- The likelihood function is

$$L\left(\boldsymbol{\theta}, \sigma_{\varepsilon}^{2}\right)=\left(2 \pi \sigma_{\varepsilon}^{2}\right)^{-n / 2} e^{-\frac{S(\boldsymbol{\theta})}{2 \sigma_{\varepsilon}^{2}}},$$

and a 'likelihood region' on $\boldsymbol{\theta}$ is a set

$$\left\{\boldsymbol{\theta} \mid L\left(\boldsymbol{\theta}, \hat{\sigma}_{\varepsilon}^{2}\right) \geq c \cdot L\left(\hat{\boldsymbol{\theta}}, \hat{\sigma}_{\varepsilon}^{2}\right)\right\}$$

for a constant c chosen for a specified coverage probability. Since

$$\frac{L\left(\boldsymbol{\theta}, \hat{\sigma}_{\varepsilon}^{2}\right)}{L\left(\hat{\boldsymbol{\theta}}, \hat{\sigma}_{\varepsilon}^{2}\right)}=e^{-\frac{\left(S(\boldsymbol{\theta})-S\left(\hat{\boldsymbol{\theta}}\right)\right)}{2 \hat{\sigma}_{\varepsilon}^{2}}},$$

we can equivalently take the region to be

$$\left\{\boldsymbol{\theta} \left| \frac{S(\boldsymbol{\theta})-S\left(\hat{\boldsymbol{\theta}}\right)}{p} \middle/ \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{n-p} \leq c^{\prime}\right.\right\},$$

and then $c^{\prime}=F_{n-p}^{p}(1-\alpha)$ gives a coverage probability of approximately α. (Here we use the usual linear approximation.) In other words, the region consists of all points $\boldsymbol{\theta}$ which are *not rejected by the likelihood ratio test*, and can be written

$$\left\{\boldsymbol{\theta} \mid S(\boldsymbol{\theta}) \leq S\left(\hat{\boldsymbol{\theta}}\right)\left(1+\frac{p}{n-p}F^{p}_{n-p}(1-\alpha)\right)\right\}.$$

(You should show that this reduces to the usual confidence ellipsoid if the model is linear.)

- The only approximation, in the likelihood region, is of the coverage probability. The shape of the region is exact, and is not forced to be an interval or ellipsoid, as in the linear approximation methods. Here is an example in which the likelihood region contains points which cannot possibly be in confidence intervals or ellipsoids. Consider the model

  $$f(\boldsymbol{\theta}, x) = \theta_1\left(1 - e^{-\theta_2 x}\right)$$

  for the biochemical oxygen demand (BOD) data set in B&W. The response variable is a measure of oxygen requirements of samples of stream water containing various organic and inorganic substances, measured after x days. Note that as

$\theta_2 \rightarrow \infty$, $f \rightarrow \theta_1$ and so $\hat{\theta}_1 \rightarrow \bar{y}$. It is thus plausible that $(\bar{y}, \infty)'$ will be in the likelihood region and the region will be 'infinite' – this cannot happen with intervals and ellipsoids.

– The point $\boldsymbol{\theta}_0 = (\bar{y} = 14.83, \infty)'$ will be in the likelihood region as long as

$$S(\boldsymbol{\theta}_0) \leq S\left(\hat{\boldsymbol{\theta}}\right)\left(1 + \frac{p}{n-p}F_{n-p}^{p}(1-\alpha)\right). \tag{9.1}$$

From the output accompanying the fit (see below) we get

$$\begin{aligned} S(\boldsymbol{\theta}_0) &= \sum (y_i - \bar{y})^2 = 107.21, \\ S\left(\hat{\boldsymbol{\theta}}\right) &= 25.99, \end{aligned}$$

and (9.1) occurs whenever $1 - \alpha \geq .94$.

[natheight=18.2056cm, natwidth=18.2056cm, height=12.628
width=14.6361cm] C:/sw50/temp/graphics/bod1$_{fig}1_{9.pdf}$

```
days = c(1,2,3,4,5,7)
```

```
oxy = c(8.3,10.3,19,16,15.6,19.8)
... fitting the model is as in the
    previous example, and results in
    the LSEs theta1, theta2 ...


# Plot S(theta) and its contours:
grid1 = seq(0,60,length=50)
grid2 = seq(0,6,length=50)
S.theta = function(theta.one,theta.two)  {
 sum((oxy-theta.one*(1-exp(-theta.two*days)))^2)
 }    # S.theta is the sum of squares function
S.mat = matrix(0,length(grid1),length(grid2))
for (i in 1:length(grid1))  {
     for (j in 1:length(grid2))
       S.mat[i,j] = S.theta(grid1[i],grid2[j])
     }
p = length(coefs); n = length(oxy)
S.max = function(conf.level)  {
     S.theta(theta1,theta2)*(1+(p/(n-p))
     *qf(conf.level,p,n-p)) }
```

```
contour(grid1,grid2,S.mat,levels=c(S.max(.95)),
  xlab="theta1",ylab="theta2")
points(theta1,theta2,pch="+")
contour(grid1,grid2,S.mat,levels=c(S.max(.80)),
  add=T, lty=4)
```

- Inferences on subsets of parameters. Suppose $\boldsymbol{\theta}' = \left(\theta_1, \theta_2, \boldsymbol{\theta}_3'\right)$ and we want a likelihood region for (θ_1, θ_2). How should $\boldsymbol{\theta}_3$ be handled?

- Easy to compute, hard to justify: The *conditional* likelihood region evaluates $\boldsymbol{\theta}_3$ at the LSE $\hat{\boldsymbol{\theta}}_3$ (i.e. all but the first two elements of $\hat{\boldsymbol{\theta}}$):
$$LR_{cond} = \left\{ \begin{array}{c} (\theta_1, \theta_2) \left| S\left(\theta_1, \theta_2, \hat{\boldsymbol{\theta}}_3\right) \leq S\left(\hat{\boldsymbol{\theta}}\right) \right. \\ \cdot \left(1 + \frac{2}{n-p} F^2_{n-p}(1-\alpha)\right) \end{array} \right\}.$$
Note that all $\binom{p}{2}$ pairs can be compared in this way with just one regression, since only $\hat{\boldsymbol{\theta}}$ needs to be computed. This region is ‘conditional’ on the event $\boldsymbol{\theta}_3 = \hat{\boldsymbol{\theta}}_3$. But the behaviour near $\hat{\boldsymbol{\theta}}_3$ may

not be representative of the behaviour elsewhere, in particular near $\hat{\hat{\boldsymbol{\theta}}}_3 = \hat{\hat{\boldsymbol{\theta}}}_3\,(\theta_1, \theta_2)$, the minimizer of S for fixed (θ_1, θ_2).

- Harder to compute but more meaningful is the *profile* likelihood region:

$$LR_{prof} = \left\{ \begin{array}{c} (\theta_1, \theta_2) \mid S\left(\theta_1, \theta_2, \hat{\hat{\boldsymbol{\theta}}}_3\right) \leq S\left(\hat{\boldsymbol{\theta}}\right) \\ \cdot \left(1 + \frac{2}{n-p} F^2_{n-p}\left(1-\alpha\right)\right) \end{array} \right\}; \quad (9.2)$$

this requires a separate regression to determine if any given pair $\left(\theta_i, \theta_j\right)$ is in the region or not. It is equivalent to placing in the likelihood region all points not rejected by the F-test, with

$$F_{obs} \overset{why?}{=} \frac{S\left(\theta_1, \theta_2, \hat{\hat{\boldsymbol{\theta}}}_3\right) - S\left(\hat{\boldsymbol{\theta}}\right)}{2} \Bigg/ \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{n-p}.$$

- Software is available to do profile regions for single parameters. The test procedure which led to (9.2) gives, in this case (with $\boldsymbol{\theta}' = \left(\theta_1, \boldsymbol{\theta}_2'\right)$)

$$\begin{aligned} LR_{prof} &= \left\{\theta_1 \mid |t_{obs}| \leq t_{n-p}\left(1 - \frac{\alpha}{2}\right)\right\}, \text{ where} \\ |t_{obs}| &= \sqrt{F_{obs}} = \sqrt{\frac{S\left(\theta_1, \hat{\hat{\boldsymbol{\theta}}}_2\right) - S\left(\hat{\boldsymbol{\theta}}\right)}{S^2}}. \end{aligned}$$

B&W define

$$\tau\left(\theta_1\right) = sign\left(\theta_1 - \hat{\theta}_1\right) \cdot \sqrt{F_{obs}}.$$

For the same reasons that likelihood regions become confidence ellipsoids in linear models, $\tau\left(\theta_1\right)$ becomes the (linear) function $\left(\theta_1 - \hat{\theta}_1\right)/se\left(\hat{\theta}_1\right)$. A plot of $\tau\left(\theta_1\right)$ against θ_1 thus indicates the degree of nonlinearity in the model.

- ‘Profiling’ code:

```
pr.bod <- profile(fit.bod)
plot(pr.bod, conf = c(95, 90, 80, 50)/100)
plot(pr.bod, conf = c(95, 90, 80, 50)/100,
    absVal = FALSE)
mtext("Confidence intervals based on the
   profile sum of squares",side = 3, outer = TRUE)
mtext("BOD data - confidence levels of 50%,
  80%, 90% and 95%",side = 1, outer = TRUE)
```

- The R code above results in the following profile plots for θ_1 (left; $\hat{\theta}_1 = 19.14$) and θ_2 (right; $\hat{\theta}_2 = .53$). Top plots have $|t_{obs}| = |\tau(\theta)|$ on vertical axis; bottom plots have $\tau(\theta)$.

[natheight=18.2056cm, natwidth=18.2056cm, height=12.815 width=12.8151cm] C:/sw50/temp/graphics/bod1$_{fig2_{10.pdf}}$

10. Lubricant data set; Starting values

- Lubricant data set from B&W: $Y = \ln(viscosity)$ of a lubricant, $x_1 = temp$ ($^\circ C$), $x_2 = pressure$, $w_2 = x_2/1000$ (so that x_1 and w_2 are of the same order). Proposed model is $Y = f(\boldsymbol{\theta}; x_1, w_2) + \varepsilon$ with

$$\begin{aligned} f(\boldsymbol{\theta}; x_1, w_2) &= \frac{\theta_1}{\theta_2 + x_1} + \theta_3 w_2 + \theta_4 w_2^2 + \theta_5 w_2^3 \\ &\quad + \left(\theta_6 + \theta_7 w_2^2\right) w_2 e^{-\frac{x_1}{\theta_8 + \theta_9 w_2^2}}. \end{aligned}$$

- Note that the model is conditionally linear, given $(\theta_2, \theta_8, \theta_9)$. We can write it as

$$Y(x_1, w_2) = \mathbf{z}'(\boldsymbol{\phi}; x_1, w_2)\boldsymbol{\beta} + \varepsilon$$

where

$$\begin{aligned} &\boldsymbol{\beta} = (\theta_1, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7), \ \boldsymbol{\phi} = (\theta_2, \theta_8, \theta_9), \\ &\mathbf{z}'(\boldsymbol{\phi}; x_1, w_2) = \\ &\left(\frac{1}{\theta_2 + x_1}, w_2, w_2^2, w_2^3, w_2 e^{-\frac{x_1}{\theta_8 + \theta_9 w_2^2}}, w_2^3 e^{-\frac{x_1}{\theta_8 + \theta_9 w_2^2}}\right). \end{aligned}$$

[natheight=18.2056cm, natwidth=18.2056cm, height=11.1764cm, width=12.1825cm]
C:/sw50/temp/graphics/luricant$_{11.pdf}$
Lubricant data and fitted regression curves.

- The 'Golub-Pereyra' algorithm requires starting values $\boldsymbol{\phi}_{(0)}$ only, and will alternate between minimizing $\sum \left(y_i - \mathbf{z}_i^{\prime} \left(\boldsymbol{\phi}_{(k)} \right) \boldsymbol{\beta} \right)^2$ by OLS to get $\boldsymbol{\beta}_{(k+1)}$, and minimizing $\sum \left(y_i - \mathbf{z}_i^{\prime} \left(\boldsymbol{\phi} \right) \boldsymbol{\beta}_{(k+1)} \right)^2$ by Gauss-Newton to get $\boldsymbol{\phi}_{(k+1)}$. In R, the call is to `nls(viscosity~cbind(...), start = starting.values,algorithm = "plinear", ...)`, where the first '...' represents the columns of $\mathbf{Z}$. This statement is preceded by `starting.values = list(theta2 = 192, theta8 = 31.73, theta9 = 0)`.

- Here is the output, which gives $SSE = .08744$; B&W report $SSE = .08996$.

```
> print(summary(fit))
Formula: viscosity ~cbind(1/(theta2 + temp),
pressure, pressure^2, pressure^3,... )

Parameters:
                Estimate Std. Error t value
theta2         2.066e+02  5.302e+00  38.968
theta8         5.743e+01  2.377e+00  24.163
theta9        -4.760e-01  7.203e-02  -6.608
.lin1          1.055e+03  2.470e+01  42.710
.lin2          1.460e+00  3.822e-02  38.200
.lin3         -2.595e-01  1.436e-02 -18.078
.lin4          2.255e-02  1.765e-03  12.781
.lin5          4.018e-01  3.363e-02  11.947
.lin6          3.527e-02  1.391e-03  25.356
---

Residual standard error:
  0.04458 on 44 degrees of freedom

> coefs = coef(fit)
> theta = c(coefs[4], coefs[1], coefs[5], coefs[6],
  coefs[7], coefs[8], coefs[9], coefs[2], coefs[3])
> names(theta) = paste("theta",1:9)
> print(theta)
      theta 1       theta 2       theta 3 ...
1054.86259672  206.61107053    1.45997825 ...
```

- Using starting values too far away from those used here results in premature termination, with the message that the gradient is singular. How were these values obtained?

1. It often helps to see what happens to the response as variables approach limiting values. Several cases in this data use the minimum value $w_2 =$ .001. As $w_2 \to 0$, $f(\boldsymbol{\theta}; x_1, w_2) \to \frac{\theta_1}{\theta_2 + x_1}$, which is transformably linear. So regress $1/y$ on x_1, using only data for which $w_2 = .001$. Obtain $\theta_{1(0)} = 984.0403$, $\theta_{2(0)} = 192.1757$.

```
fit1 = lsfit(x1[w2==.001], 1/y[w2==.001])$coef
```

```
t1 = 1/fit1[2]
t2 = t1*fit1[1]
cat("t1 =", t1, "t2 =", t2, "\n")
t1 = 984.0403 t2 = 192.1757
```

2. For small w_2 we can possibly ignore terms of order 2 and higher in w_2:

$$f\left(\boldsymbol{\theta}; x_1, w_2\right) \approx \frac{\theta_1}{\theta_2 + x_1} + w_2 \left(\theta_3 + \theta_6 e^{-\frac{x_1}{\theta_8}}\right). \tag{10.1}$$

   We write this as

$$y - \frac{\theta_{1(0)}}{\theta_{2(0)} + x_1} \approx w_2 \beta$$

   and regress $u = y - \frac{\theta_{1(0)}}{\theta_{2(0)}+x_1}$ on w_2, using only values for which $w_2 < 2$. We do this for each temperature group, i.e. each value of x_1.

```
 (y.1, w2.1, x1.1 refer to the first temp.\ group)
u.1 = y.1 - t1/(t2+x1.1)
beta.1 =
  lsfit(w2.1[w2.1<2], u.1[w2.1<2], int=F)$coef
... three more of these ...
 cat(...)
betas are 1.573094 1.484346 1.390471 1.362453
```

We obtain

$$\begin{array}{lcccc} x_1: & 0 & 25 & 37.8 & 98.9 \\ \hat{\beta}: & 1.573094 & 1.484346 & 1.390471 & 1.362453 \end{array}.$$

Now $x_1 \to 0 \Rightarrow \beta \to \theta_3 + \theta_6$ and $x_1 \to \infty \Rightarrow \beta \to \theta_3$. So we take

$$\begin{aligned} \theta_{3(0)} &= 1.35, \\ \theta_{6(0)} &= 1.57 - \theta_{3(0)} = .22. \end{aligned}$$

Then, since

$$x_1 = \theta_8 \log\left(\frac{\theta_6}{\beta - \theta_3}\right)$$

we regress the 4 values of x_1 on the 4 values of $\log\left(\frac{\theta_{6(0)}}{\beta - \theta_{3(0)}}\right)$ to get $\theta_{8(0)} = 31.72969$.

```
t3 = 1.35
```

```
t6 = .22
v = log(t6/(c(beta.1,beta.2,beta.3,beta.4) - t3))
t8 = lsfit(v, unique(x1), int = F)$coef
cat("t8 = ", t8, "\n")
t8 =  31.72969
```

3. B&W refine these initial values further by doing a full (Golub-Pereyra) nonlinear regression in model (10.1), using $w_2 < 2$ and the starting values $\theta_{2(0)}$ and $\theta_{8(0)}$ already obtained, to get refined values $\theta_{2(0)} = 202$, $\theta_{8(0)} = 35.9$. They then set $\theta_9 = 0$ and use all of the data (but model (10.1) still?) and Golub-Pereyra again, so that only $\theta_{2(0)}$ and $\theta_{8(0)}$ need to be specified, to get $\theta_{2(0)} = 209$, $\theta_{8(0)} = 47.6$. Finally, they use these starting values together with $\theta_{9(0)} = 0$ to estimate the full model. However, our starting values (including $\theta_{9(0)} = 0$) are evidently close enough.

11. Hypothesis testing

- First consider testing the entire parameter vector: *H*: $\boldsymbol{\theta} = \boldsymbol{\theta}_0$. There are two common options. The first is derived from the linear approximation confidence ellipsoid on $\boldsymbol{\theta}$, and prescribes that we reject those values not contained in the ellipsoid. This is also called 'Wald's test'. The p-value is $P\left(F^p_{n-p} > F_1\right)$, where

$$F_1 = \frac{\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\right)' \left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right) \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\right)}{pS^2}$$

and under *H*, $F_1 \overset{d}{\approx} F^p_{n-p}$. Here $\hat{\mathbf{V}} = \mathbf{V}\left(\hat{\boldsymbol{\theta}}\right)$.

  - One drawback here is that the approximation of the distribution as F^p_{n-p} might be quite poor if the nonlinearity is severe.

  - More striking is that F_1 is not invariant under reparameterizations. For instance in the model $f\left(\theta_0, \theta_1; x\right) = e^{\theta_0 + \theta_1 x}$ a test of *H*: $\theta_0 = 0, \theta_1 = 1$ should result in the same conclusion if we reparameterize as $f(\phi_0, \theta_1; x) = \phi_0 e^{\theta_1 x}$ ($\phi_0 = e^{\theta_0}$) and test *H*: $\phi_0 = 1, \theta_1 = 1$. The two values of F_1 will however be different.

– In general, suppose we start with a model $\mathbf{Y} = \boldsymbol{\eta}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}$, and propose to test *H*: $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ using a test statistic $F(\boldsymbol{\theta}_0)$. Suppose we reparameterize by introducing a $1-1$, differentiable map $\mathrm{g} : \boldsymbol{\theta} \rightarrow \boldsymbol{\phi} = \mathrm{g}(\boldsymbol{\theta})$. Define $\tilde{\boldsymbol{\eta}}(\boldsymbol{\phi}) = \boldsymbol{\eta}\left(\mathrm{g}^{-1}(\boldsymbol{\phi})\right)$ so that the model is $\mathbf{Y} = \tilde{\boldsymbol{\eta}}(\boldsymbol{\phi}) + \boldsymbol{\varepsilon}$ and we test *H*: $\boldsymbol{\phi} = \boldsymbol{\phi}_0$ where $\boldsymbol{\phi}_0 = \mathrm{g}(\boldsymbol{\theta}_0)$. The test is *invariant* if $F(\boldsymbol{\phi}_0) = F(\boldsymbol{\theta}_0)$.

- The test based on likelihood regions, rather than confidence ellipsoids, is typically less affected by curvature and is invariant. It uses

$$F_2 = \frac{S(\boldsymbol{\theta}_0) - S\left(\hat{\boldsymbol{\theta}}\right)}{p} \Bigg/ \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{n-p}.$$

Again, under *H*, $F_2 \overset{d}{\approx} F^p_{n-p}$. This approximate distribution will be derived later. For the invariance note that

$$F_2\left(\phi_0\right) = \frac{\tilde{S}\left(\phi_0\right) - \tilde{S}\left(\hat{\phi}\right)}{p} \Bigg/ \frac{\tilde{S}\left(\hat{\phi}\right)}{n-p}\,.$$

where

$$\begin{aligned} \tilde{S}\left(\phi\right) &= \left\|\mathbf{y} - \tilde{\boldsymbol{\eta}}\left(\phi\right)\right\|^2 \\ &= \left\|\mathbf{y} - \boldsymbol{\eta}\left(\mathbf{g}^{-1}\left(\phi\right)\right)\right\|^2 = S\left(\mathbf{g}^{-1}\left(\phi\right)\right). \end{aligned}$$

Thus

$$\tilde{S}\left(\phi_0\right) = S\left(\mathbf{g}^{-1}\left(\phi_0\right)\right) = S\left(\boldsymbol{\theta}_0\right)$$

and

$$\begin{aligned} \tilde{S}\left(\hat{\phi}\right) &= \min_{\phi} \tilde{S}\left(\phi\right) = \min_{\phi} S\left(\mathbf{g}^{-1}\left(\phi\right)\right) \\ &= \min_{\boldsymbol{\theta}} S\left(\boldsymbol{\theta}\right) = S\left(\hat{\boldsymbol{\theta}}\right), \end{aligned}$$

since $\left\{\mathbf{g}^{-1}\left(\phi\right) \mid \phi \in \mathbf{R}^p\right\} = \left\{\boldsymbol{\theta} \mid \boldsymbol{\theta} \in \mathbf{R}^p\right\}$. Furthermore $\hat{\phi} = \mathbf{g}\left(\hat{\boldsymbol{\theta}}\right)$.

- More common is to test single parameters, or subsets of the parameter vector. Suppose
$$\boldsymbol{\theta} = \begin{pmatrix} \boldsymbol{\theta}_1 \\ \boldsymbol{\theta}_2 \end{pmatrix} \begin{matrix} p_1 \\ p_2 = p - p_1 \end{matrix}$$
and we test *H*: $\boldsymbol{\theta}_2 = \boldsymbol{\theta}_{2,0}$. The linear approximation F-test is based on the approximate normality of the LSE:
$$\hat{\boldsymbol{\theta}} - \boldsymbol{\theta} \overset{d}{\approx} N\left(0, \sigma_\varepsilon^2 \left(\mathbf{V}'\mathbf{V}\right)^{-1}\right).$$
Partition the covariance matrix as
$$\left(\mathbf{V}'\mathbf{V}\right)^{-1} = \begin{pmatrix} (\mathbf{V}'\mathbf{V})^{11} & (\mathbf{V}'\mathbf{V})^{12} \\ (\mathbf{V}'\mathbf{V})^{21} & (\mathbf{V}'\mathbf{V})^{22} \end{pmatrix}.$$
Then under *H*,
$$\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta}_{2,0} \overset{d}{\approx} N\left(0, \sigma_\varepsilon^2 \left(\mathbf{V}'\mathbf{V}\right)^{22}\right)$$
and so
$$F_1 = \frac{\left(\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta}_{2,0}\right)' \left[\left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right)^{22}\right]^{-1} \left(\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta}_{2,0}\right)}{p_2 S^2} \overset{d}{\approx} F_{n-p}^{p_2}.$$
To compute, the QR-decomposition is useful. (This

is a by-product of `lsfit(...)`.) If

$$\hat{\mathbf{V}} = \mathbf{Q}_1\mathbf{R}_1 \text{ and } \mathbf{R}_1 = \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ 0 & \mathbf{R}_{22} \end{pmatrix}$$

then $\left[\left(\hat{\mathbf{V}}'\hat{\mathbf{V}}\right)^{22}\right]^{-1} = \cdots = \mathbf{R}_{22}'\mathbf{R}_{22}$ and so

$$F_1 = \frac{\left\|\mathbf{R}_{22}\left(\hat{\boldsymbol{\theta}}_2 - \boldsymbol{\theta}_{2,0}\right)\right\|^2}{p_2 S^2}.$$

- This test statistic suffers from the same problems as when the entire parameter vector is being tested. Simulations indicate that the effect of curvature is even more severe when p_2 is small relative to p.

- The likelihood ratio test rejects for large values of

$$F_2 = \frac{S\left(\hat{\hat{\boldsymbol{\theta}}}_1, \boldsymbol{\theta}_{2,0}\right) - S\left(\hat{\boldsymbol{\theta}}\right)}{p_2} \Bigg/ \frac{S\left(\hat{\boldsymbol{\theta}}\right)}{n-p}.$$

This is invariant under reparameterizations. To obtain the approximate distribution, write $\boldsymbol{\theta}_{1,0}$ for the *true* value of $\boldsymbol{\theta}_1$ and

$$\begin{aligned}
& S\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_{2,0}\right) \\
= \; & \|\mathbf{y} - \boldsymbol{\eta}_1(\boldsymbol{\theta}_1)\|^2 \text{ for } \boldsymbol{\eta}_1(\boldsymbol{\theta}_1) = \boldsymbol{\eta}\begin{pmatrix} \boldsymbol{\theta}_1 \\ \boldsymbol{\theta}_{2,0} \end{pmatrix} \\
= \; & \left\|\mathbf{y} - \boldsymbol{\eta}_1\left(\boldsymbol{\theta}_{1,0}\right) - \left(\boldsymbol{\eta}_1(\boldsymbol{\theta}_1) - \boldsymbol{\eta}_1\left(\boldsymbol{\theta}_{1,0}\right)\right)\right\|^2 \\
\approx \; & \left\|\boldsymbol{\varepsilon} - \mathbf{V}_{1,0}\left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_{1,0}\right)\right\|^2,
\end{aligned}$$

where

$$\mathbf{V}_{1,0} = \frac{\partial \boldsymbol{\eta}_1}{\partial \boldsymbol{\theta}_1}|_{\boldsymbol{\theta}_{1,0}, \boldsymbol{\theta}_{2,0}}$$

is the first p_1 columns of $\mathbf{V}_0 = \mathbf{V}\left(\boldsymbol{\theta}_{1,0}, \boldsymbol{\theta}_{2,0}\right)$. To this order of approximation, $S\left(\boldsymbol{\theta}_1, \boldsymbol{\theta}_{2,0}\right)$ is minimized by

$$\hat{\hat{\boldsymbol{\theta}}}_1 = \boldsymbol{\theta}_{1,0} + \left(\mathbf{V}_{1,0}'\mathbf{V}_{1,0}\right)^{-1}\mathbf{V}_{1,0}'\boldsymbol{\varepsilon}$$

with minimum value

$$\begin{aligned}
S\left(\hat{\hat{\boldsymbol{\theta}}}_1, \boldsymbol{\theta}_{2,0}\right) &= \boldsymbol{\varepsilon}'\left(\mathbf{I} - \mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon} \text{ for} \\
\mathbf{H}_{1,0} &= \mathbf{V}_{1,0}\left(\mathbf{V}_{1,0}'\mathbf{V}_{1,0}\right)^{-1}\mathbf{V}_{1,0}'.
\end{aligned}$$

Using the same approximations,

$$S\left(\hat{\boldsymbol{\theta}}\right) = \boldsymbol{\varepsilon}'(\mathbf{I} - \mathbf{H}_0)\boldsymbol{\varepsilon} \text{ for } \mathbf{H}_0 = \mathbf{V}_0\left(\mathbf{V}_0'\mathbf{V}_0\right)^{-1}\mathbf{V}_0'.$$

Thus

$$F_2 \approx \frac{\boldsymbol{\varepsilon}'\left(\mathbf{H}_0 - \mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon}}{p_2} \Big/ \frac{\boldsymbol{\varepsilon}'\left(\mathbf{I} - \mathbf{H}_0\right)\boldsymbol{\varepsilon}}{n-p}.$$

Now apply Gram-Schmidt to get the distribution. First let the columns of $\mathbf{Q}_1 : n \times p_1$ be an orthonormal basis for col$\left(\mathbf{V}_{1,0}\right)$, so that $\mathbf{H}_{1,0} = \mathbf{Q}_1\mathbf{Q}_1'$. Extend this to an orthonormal basis for col$\left(\mathbf{V}_0\right)$ consisting of the columns of $(\mathbf{Q}_1 \vdots \mathbf{Q}_2)$, so that $\mathbf{H}_0 - \mathbf{H}_{1,0} = \mathbf{Q}_2\mathbf{Q}_2'$ where $\mathbf{Q}_2$ is $n \times p_2$. Finally extend to an orthogonal matrix $(\mathbf{Q}_1 \vdots \mathbf{Q}_2 \vdots \mathbf{Q}_3)$, so that $\mathbf{I} - \mathbf{H}_0 = \mathbf{Q}_3\mathbf{Q}_3'$ where $\mathbf{Q}_3$ is $n \times (n-p)$. Then

$$F_2 \approx \frac{\left\|\mathbf{Q}_2'\boldsymbol{\varepsilon}\right\|^2}{p_2} \Big/ \frac{\left\|\mathbf{Q}_3'\boldsymbol{\varepsilon}\right\|^2}{n-p} \sim F_{n-p}^{p_2}$$

since

$$\binom{\mathbf{Q}_2'\boldsymbol{\varepsilon}}{\mathbf{Q}_3'\boldsymbol{\varepsilon}} \sim N\left(\binom{\mathbf{0}}{\mathbf{0}}, \sigma_\varepsilon^2 \mathbf{I}_{n-p_1}\right),$$

implying that $\left\|\mathbf{Q}_2'\boldsymbol{\varepsilon}\right\|^2 \sim \sigma_\varepsilon^2\chi_{p_2}^2$, independently of $\left\|\mathbf{Q}_3'\boldsymbol{\varepsilon}\right\|^2 \sim \sigma_\varepsilon^2\chi_{n-p}^2$.

- In the notation

$$\begin{aligned}\hat{\boldsymbol{\varepsilon}} &= \mathbf{y} - \boldsymbol{\eta}\begin{pmatrix}\hat{\boldsymbol{\theta}}_1 \\ \hat{\boldsymbol{\theta}}_2\end{pmatrix} \approx (\mathbf{I} - \mathbf{H}_0)\,\boldsymbol{\varepsilon}, \\ \hat{\hat{\boldsymbol{\varepsilon}}} &= \mathbf{y} - \boldsymbol{\eta}\begin{pmatrix}\hat{\hat{\boldsymbol{\theta}}}_1 \\ \boldsymbol{\theta}_{2,0}\end{pmatrix} \approx \left(\mathbf{I} - \mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon}\end{aligned}$$

we have

$$F_2 = \frac{\left\|\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2 - \left\|\hat{\boldsymbol{\varepsilon}}\right\|^2}{p_2}\Big/ S^2\,.$$

Two other proposals in the literature are

$$F_3 = \frac{\left\|\hat{\hat{\mathbf{H}}}\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2}{p_2}\Big/ S^2\,,$$

where $\hat{\hat{\mathbf{H}}} = \hat{\hat{\mathbf{V}}}\left(\hat{\hat{\mathbf{V}}}'\hat{\hat{\mathbf{V}}}\right)^{-1}\hat{\hat{\mathbf{V}}}'$ and $\hat{\hat{\mathbf{V}}} = \mathbf{V}\left(\hat{\hat{\boldsymbol{\theta}}}_1, \boldsymbol{\theta}_{2,0}\right)$, and

$$F_4 = \frac{\left\|\hat{\hat{\mathbf{H}}}\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2}{p_2}\Bigg/ \frac{\left\|\left(\mathbf{I} - \hat{\hat{\mathbf{H}}}\right)\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2}{n-p}\,.$$

Both are $\overset{d}{\approx} F^{p_2}_{n-p}$; F_4 has the computational advantage of not requiring $\hat{\boldsymbol{\theta}}$. Simulations indicate that F_2 is typically more powerful than F_3 , which in turn is typically more powerful than F_4.

- Motivation for F_3 is that the numerator is asymptotically equivalent to that of F_2: As $n \to \infty$, $\left(\hat{\hat{\boldsymbol{\theta}}}_1, \boldsymbol{\theta}_{2,0}\right)$ converges to $\left(\boldsymbol{\theta}_{1,0}, \boldsymbol{\theta}_{2,0}\right)$, hence $\hat{\hat{\mathbf{H}}}$ to $\mathbf{H}_0$ and $\left\|\hat{\hat{\mathbf{H}}}\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2$ to $\left\|\mathbf{H}_0\left(\mathbf{I}-\mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon}\right\|^2 = \left\|\left(\mathbf{H}_0-\mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon}\right\|^2$ (since $\mathbf{H}_0\mathbf{v} = \mathbf{v}$ for any $\mathbf{v} \in$ col $(\mathbf{V}_0)$ and the columns of $\mathbf{H}_{1,0}$ are in col $\left(\mathbf{V}_{1,0}\right) \subset$ col $(\mathbf{V}_0)$).

- The motivation for F_4 now follows – its denominator is
$$\frac{\left\|\left(\mathbf{I}-\hat{\hat{\mathbf{H}}}\right)\hat{\hat{\boldsymbol{\varepsilon}}}\right\|^2}{n-p} \approx \frac{\left\|\left(\mathbf{I}-\mathbf{H}_0\right)\left(\mathbf{I}-\mathbf{H}_{1,0}\right)\boldsymbol{\varepsilon}\right\|^2}{n-p} = S^2,$$
since $\left(\mathbf{I}-\mathbf{H}_0\right)\mathbf{H}_{1,0} = 0$.

Part III

Smoothing; Alternatives to Least Squares

12. Splines and other bases

- Good reading material for these lectures:

  1. Venables & Ripley; chapter on ‘Nonlinear and Smooth Regression’ or on ‘Modern regression’, depending on the edition.

  2. Hastie, Tibshirani & Friedman, ch. 5, 6, 9.

- Observe $y_i = f(\mathbf{x}_i) + \varepsilon_i$ but no knowledge of $f(\cdot)$; determine $\hat{f}(\mathbf{x})$ from the data alone – no model.

- Output from these methods is typically graphical and used for prediction and interpolation. The distribution theory needed for inferences is as yet largely undeveloped.

- The ‘motorcycle’ data gives measurements on head acceleration vs. milliseconds after impact in a simulated motorcycle accident; it is used to test crash helmets.

[natheight=18.2056cm, natwidth=18.2056cm, height=12.182
width=12.1825cm] C:/sw50/temp/graphics/mcycle1$_{fig}1_{12.pd}$

- One might try to fit a linear combination of certain ‘basis’ functions, such as orthogonal polynomials (see `help(poly)`):

[natheight=18.2056cm, natwidth=18.2056cm, height=10.926
width=12.2681cm] C:/sw50/temp/graphics/mcycle1$_{fig}2_{13.pd}$

- Polynomials can be very unstable to fit, and behave erratically away from the region where there are data.

- Fourier series:

$$\hat{y}(t) = \hat{\theta}_0 + \sum_{k=1}^{K} \left(\hat{\theta}_{2k-1} \sin\left(k\omega t\right) + \hat{\theta}_{2k} \cos\left(k\omega t\right) \right);$$

  ω chosen so that the period $2\pi/\omega$ is the range of the data. Good for approximating very smooth functions with no strong local features and the same degree of curvature everywhere.

[natheight=18.2056cm, natwidth=18.2056cm, height=11.176
width=12.1825cm] C:/sw50/temp/graphics/mcycle1$_{f}ig3_{14.pd}$

- Splines. Suppose we wish to approximate a function $f\left(x\right)$ over an interval $[x_1, x_N]$, and require that the approximating function $s(x)$ satisfy:

  1. For given ‘nodes’ $x_1 < x_2 < \cdots < x_{N-1} < x_N$, $s(x)$ is a cubic polynomial on each interval $[x_i, x_{i+1}]$;

  2. $s(x_i) = f(x_i)$ at each node;

3. The second (hence the first) derivative $s''(x)$ exists and is continuous throughout $[x_1, x_N]$;

4. $s''(x_1) = s''(x_N) = 0$.

There is exactly one such function satisfying these properties. (Outline of proof: A cubic function is determined by 4 parameters on each interval; one shows that the available parameters are uniquely determined by the system of linear equations implied by (2) - (4). See http://mathworld.wolfram.com/CubicSpline.html for instance.)

The solution (taken to be linear outside of $[x_1, x_N]$) is called the 'natural cubic spline'. If $g(x)$ is any other twice continuously differentiable function (we write $g \in C^2[x_1, x_N]$) interpolating $f(x)$ at the nodes (i.e. satisfying $g(x_i) = f(x_i)$ at each node) then

$$\int_{x_1}^{x_N} \left[g''(x)\right]^2 dx \geq \int_{x_1}^{x_N} \left[s''(x)\right]^2 dx,$$

with equality iff $g(x) \equiv s(x)$ on $[x_1, x_N]$ (assigned).

Suppose now that we have data $y = y(x)$ and are to solve the 'penalized regression' problem

$$\min_{g \in C^2[x_1, x_N]} \left\{ \sum_{i=1}^{N} (y_i - g(x_i))^2 + \lambda \int_{x_1}^{x_N} \left[g''(x) \right]^2 dx \right\}, \tag{12.1}$$

for a *smoothing parameter* $\lambda > 0$. ($\lambda = 0 \Rightarrow ?$; $\lambda = \infty \Rightarrow ?$.) By the preceding, the solution is a cubic spline; this is because, for any candidate $g(x)$, the smoothing spline interpolating $g(x)$ at the nodes has the same SS and a smaller penalty. So we can restrict the search to splines. Having learned this we drop requirement 2. The nodes might be taken to be N ($\leq$ # of unique x-values) equally spaced values of x in the data, or perhaps all unique values of x. One can then represent the spline as a linear combination of basis elements

$$s(x) = \sum_{j=1}^{N} \theta_j b_j(x) = \mathbf{b}'(x)\boldsymbol{\theta}$$

in a variety of ways. One is

$$\begin{aligned} b_1(x) &= 1, \; b_2(x) = x, \\ b_{j+2}(x) &= d_j(x) - d_{N-1}(x), \; j = 1, ..., N-2, \end{aligned}$$

where

$$d_j(x) = \frac{\left(x - x_j\right)_+^3 - (x - x_N)_+^3}{x_N - x_j}.$$

You should check the continuity of b_j, b_j' and b_j''; that of b_j and b_j' is inherited from the d_j. More sophisticated is a 'B-spline' basis. Once a basis and knot sequence are chosen, (12.1) becomes a parametric problem:

$$\min_{\boldsymbol{\theta}} \left\{ \|\mathbf{y} - \mathbf{L}\boldsymbol{\theta}\|^2 + \lambda \boldsymbol{\theta}' \mathbf{G} \boldsymbol{\theta} \right\}$$

where $\mathbf{L}$ has rows $\mathbf{b}'(x_i)$ and, where $\ddot{\mathbf{b}}$ is the vector of second derivatives, $\mathbf{G} = \int_{x_1}^{x_N} \ddot{\mathbf{b}}(x)\ddot{\mathbf{b}}'(x)dx \geq \mathbf{0}$. Similar to ridge regression (but here $\mathbf{L}_{N \times N}$ is square),

$$\hat{\boldsymbol{\theta}} = \left[\mathbf{L}'\mathbf{L} + \lambda \mathbf{G}\right]^{-1} \mathbf{L}'\mathbf{y} \text{ and } \hat{\mathbf{y}} {=} \mathbf{L}\hat{\boldsymbol{\theta}} = \mathbf{S}_\lambda \mathbf{y},$$

where the 'smoother' matrix $\mathbf{S}_\lambda$ plays the same role as the hat matrix. The *equivalent degrees of freedom* (or *equivalent number of parameters*) are thus

$$df_\lambda = tr\left[\mathbf{S}_\lambda\right].$$

```
fit = smooth.spline(times, accel)
plot(mcycle)
lines(fit, col=1, lty=1)
lines(smooth.spline(times, accel, df=2), col=2)
lines(smooth.spline(times, accel, df=5), col=4)
lines(smooth.spline(times, accel, df=60), col=6)
legend("bottomright", legend = c("df=12.21 (GCV)",
 "df=2", "df=5", "df=60"), col=c(1,2,4,6))
```

[natheight=17.6279cm, natwidth=20.3276cm, height=11.527
width=13.2852cm] C:/sw50/temp/graphics/mcycle1$_{f}ig4_{1}5.pd$

```
> fit
smooth.spline(x = times, y = accel)
Smoothing Parameter  spar= 0.6598558
lambda= 0.00011075 (14 iterations)
Equivalent Degrees of Freedom (Df): 12.20876
Penalized Criterion: 38650.54
GCV: 565.4513
```

- The smoothness is controlled by λ; it and df_λ can be determined from each other. The R function will determine an optimal λ_0 by 'generalized cross-validation':

  $$\lambda_0 = \arg\min_\lambda \frac{\left\|(\mathbf{I} - \mathbf{S}_\lambda)\,\mathbf{y}\right\|^2}{n - df_\lambda}.$$

  This is the default; an option is ordinary cross-validation:

  $$\lambda_0 = \arg\min_\lambda \sum_{i=1}^{n} \left(\frac{e_i(\lambda)}{1 - [\mathbf{S}_\lambda]_{ii}} \right)^2.$$

  This is derived from a more general principle – in order to determine the best terms to include in a model we might take a proposed model, leave out one observation, fit the model and see how well it predicts the omitted observation (an overparameterized model will not do this well):

  $$PRESS = \sum_{i=1}^{n} \left(y_i - \hat{y}_{i(i)} \right)^2$$

  as at p. 74, Lecture 7. Choose the model which minimizes this 'Prediction Error Sum of Squares'.

13. Kernel Smoothing; Local Regression

- When there is no parametric model relating the fitted values at one point to those at other points, it is reasonable to let the fit at x be determined by those points $\left(x_j, y_j\right)$ with x_j close to x. A first attempt might be ‘running means’, in which $\hat{y}_i$ is the average of the y_j with $|i-j| \leq k$ (assuming that $\cdots x_j \leq x_{j+1} \cdots$). Alternatively, ‘running medians’. (Then `plot(...,type="l")` for linear interpolation between the $(x_i, \hat{y}_i)$.)

```
runningmean = function(k) {
runm = rep(0, n)
for(i in (k+1):(n-k))
 runm[i] = mean(accel[(i-k):(i+k)])
runm = runm[(k+1):(n-k)]
}
```

[natheight=18.2056cm, natwidth=18.2056cm, height=13.428cm, width=14.6361cm]
C:/sw50/temp/graphics/mcycle2$_{f}ig1_{16.pdf}$
Running means (top) and medians (bottom).

The ‘super smoother’ function `supsmu(...)` on R will replace running means with running linear regressions – at each point (x_i, y_i), $\hat{y}_i$ is obtained by doing a linear regression using only k nearby points as data. ($+$ sophisticated modifications – see the R help file.)

[natheight=18.2056cm, natwidth=18.2056cm, height=10.926cm, width=11.9035cm]
C:/sw50/temp/graphics/mcycle2$_fig1a_{1}7.pdf$
Super smooth fit (`supsmu(times, accel, bass=?)`) to motorcycle data; bass = 0 is default. The arguments ‘span’ ($= k/n$) can be chosen by cross-validation (the default) or specified.

- More flexible is ‘kernel smoothing’, in which the fitted value at x is a weighted average of those values of y observed at points x_j near x:

$$\hat{y}(x) = \sum_{i=1}^{n} w\left(x - x_i\right) y_i,$$

where $w(x-x_i)$ is typically a symmetric function, decreasing in $|x-x_i|$ and satisfying

$$\sum_{i=1}^{n} w\left(x-x_i\right)=1.$$

The ‘Nadaraya-Watson’ kernel uses

$$w(x-x_i)=\frac{K_\lambda\left(x-x_i\right)}{\sum_{j=1}^{n} K_\lambda\left(x-x_j\right)},$$

where $K(t)$ is a unimodal probability density, symmetric about 0, and $K_\lambda(t)=\frac{1}{\lambda}K\left(\frac{t}{\lambda}\right)$. (So $\lambda\longrightarrow 0 \Rightarrow \hat{y}(x)\longrightarrow ?$; $\lambda \longrightarrow \infty \Rightarrow \hat{y}(x)\longrightarrow ?$)

- Common choices of kernel functions:

  1. Epanechnikov kernel: $K(t) = \frac{3}{4}\left(1 - t^2\right) I(|t| \leq 1)$.

  2. Tri-cube function: $K(t) \propto \left(1 - |t|^3\right)^3 I(|t| \leq 1)$.

  3. Uniform ('box' in R): $K(t) = .5 I(|t| \leq 1)$.

  4. Gaussian: $K(t) = \phi(t)$.

In R one can choose a 'bandwidth' ($= 4\cdot$ upper quartile of $K_\lambda = 4\lambda\cdot$ upper quartile of K_1):

$$.75 = \int_{-\infty}^{\text{bandwidth}/4} K_\lambda(x)dx.$$

```
plot(mcycle)
lines(ksmooth(times, accel, kernel = "box",
bandwidth = ??), lty=4)
   ... other bandwidths ...
   ... ditto, with kernel = "normal"
```

[natheight=18.2056cm, natwidth=18.2056cm, height=16.769cm, width=18.2759cm]
C:/sw50/temp/graphics/mcycle2$_{fig}2_{18.pdf}$
Kernel smooths to motorcycle data; ‘box’ kernel.
Bandwidth = .5 is the default.

[natheight=18.2056cm, natwidth=18.2056cm, height=17.2699cm, width=18.2759cm]
C:/sw50/temp/graphics/mcycle2$_{fig}3_{19.pdf}$
Kernel smooths to motorcycle data; ‘normal’ kernel.
Bandwidth = .5 is the default.

- Kernel smooths can be badly biased near the edges of the region containing the x’s (since there are too few x_i’s on one side of x). Without special conditions on the ‘design’ (the choice of the x_i) or on the kernel, they can be badly biased elsewhere. In recent years attention seems to have shifted away from kernel smoothing and towards ‘local regression’ methods.

- Example of 'local regression'. Suppose we have data $(x_i, y_i = f(x_i) + \varepsilon_i)$. For an arbitrary x_0, consider estimating $f(x_0)$ by a constant $\hat{\theta}(x_0)$ defined by

$$\begin{aligned} \hat{\theta}(x_0) &= \arg\min_{\theta} \sum_{i=1}^{n} K_\lambda(x_0 - x_i)(y_i - \theta)^2 \qquad (13.1) \\ &= \sum_{i=1}^{n} \left\{ K_\lambda(x_0 - x_i) \Big/ \sum_{j=1}^{n} K_\lambda\left(x_0 - x_j\right) \right\} y_i. \end{aligned}$$

  Thus Nadaraya-Watson kernel smoothing arises from (13.1), which we can generalize to local regression as follows. A 'local (linear) regression' has

$$\begin{aligned} \hat{f}(x_0) &= \hat{\theta}_0(x_0) + \hat{\theta}_1(x_0)\, x_0 \text{ for} \\ \hat{\boldsymbol{\theta}}(x_0) &= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} K_\lambda(x_0 - x_i)(y_i - \theta_0 - \theta_1 x_i)^2; \end{aligned}$$

  a 'locally quadratic' fit includes $\theta_2 x_i^2$, etc.

- For general multiple regression with regressors $\mathbf{x}$ one solves

$$\hat{\boldsymbol{\theta}}(\mathbf{x}_0) = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} K_\lambda(\mathbf{x}_0, \mathbf{x}_i)\left(y_i - \left(1, \mathbf{x}_i'\right)\boldsymbol{\theta}\right)^2$$

and sets $\hat{f}(\mathbf{x}_0) = \left(1, \mathbf{x}_0'\right)\hat{\boldsymbol{\theta}}(\mathbf{x}_0)$; $K_\lambda(\mathbf{x}_0, \mathbf{x}_i)$ is typically ‘radially symmetric’, i.e. a function of $\|\mathbf{x}_0 - \mathbf{x}_i\|$ such as $\frac{1}{\lambda}\phi\left(\frac{\|\mathbf{x}_0 - \mathbf{x}_i\|}{\lambda}\right)$.

- Extensions to nonlinear regression are obvious.

- An advantage of local regression estimators over kernel smoothing is in bias reduction. Consider local polynomial regression of degree r:

  $$\begin{aligned} \mathbf{z}'(x) &= (1, x, \cdots, x^r), \\ \hat{f}(x_0) &= \mathbf{z}'(x_0)\hat{\boldsymbol{\theta}}(x_0), \\ \hat{\boldsymbol{\theta}}(x_0) &= \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{n} K_\lambda(x_0, x_i)\left(y_i - \mathbf{z}'(x_i)\hat{\boldsymbol{\theta}}\right)^2. \end{aligned}$$

  Let $\mathbf{W}(x_0)$ be the diagonal matrix having the $K_\lambda(x_0, x_i)$ on its diagonal, and let $\mathbf{Z}$ be the design matrix with rows $\mathbf{z}'(x_i)$. Then from the theory of WLS estimation we get

  $$\begin{aligned} \hat{\boldsymbol{\theta}}(x_0) &= \left(\mathbf{Z}'\mathbf{W}(x_0)\mathbf{Z}\right)^{-1}\mathbf{Z}'\mathbf{W}(x_0)\mathbf{y}, \\ \hat{f}(x_0) &= \mathbf{z}'(x_0)\hat{\boldsymbol{\theta}}(x_0) = \mathbf{b}'(x_0)\mathbf{y}, \text{ for} \\ \mathbf{b}'(x_0) &= \mathbf{z}'(x_0)\left(\mathbf{Z}'\mathbf{W}(x_0)\mathbf{Z}\right)^{-1}\mathbf{Z}'\mathbf{W}(x_0). \end{aligned}$$

- We have

$$E\left[\hat{f}(x_0)\right] = \mathbf{b}'(x_0)\,E[\mathbf{y}] = \sum_{i=1}^{n} b_i(x_0)\,f(x_i)$$

$$= \sum_{i=1}^{n} b_i(x_0)\left[\begin{array}{c} f(x_0) + f'(x_0)(x_i - x_0) + \\ \cdots + f^{(r)}(x_0)\frac{(x_i - x_0)^r}{r!} + R_i \end{array}\right] \quad (13.2)$$

where the remainders R_i involve derivatives of f of higher order than r; the experimenter assumes that the curvature of f is such that these can safely be ignored. **Claim:**

$$\sum_{i=1}^{n} b_i(x_0)(x_i - x_0)^k = I(k = 0),$$

so (13.2) becomes

$$E\left[\hat{f}(x_0)\right] = f(x_0) + \sum_{i=1}^{n} b_i(x_0)\,R_i$$

and the bias involves only (one hopes) negligible curvature.

- Verification of claim: From

$$\mathbf{b}'(x_0)\,\mathbf{Z} = \mathbf{z}'(x_0)$$

we get

$$\sum_{i=1}^{n} b_i(x_0)\, x_i^j = x_0^j,\ \ j = 0, ..., r.$$

Thus for $k = 0, ..., r$,

$$\begin{aligned}
& \sum_{i=1}^{n} b_i(x_0)\,(x_i - x_0)^k \\
= \;& \sum_{i=1}^{n} b_i(x_0) \sum_{j=0}^{k} \binom{k}{j} x_i^j\,(-x_0)^{k-j} \\
= \;& \sum_{j=0}^{k} \binom{k}{j} (-x_0)^{k-j} \sum_{i=1}^{n} b_i(x_0)\, x_i^j \\
= \;& \sum_{j=0}^{k} \binom{k}{j} (-x_0)^{k-j}\, x_0^j \\
= \;& (x_0 - x_0)^k \\
= \;& I(k = 0).
\end{aligned}$$

- The variance ($\text{var}\left[\hat{f}\left(x_0\right)\right] = \sigma_\varepsilon^2 \left\|\mathbf{b}\left(x_0\right)\right\|^2$) increases as more terms are added; there is a trade-off between bias and variance.

- We have $\hat{\mathbf{y}} = \mathbf{S}_\lambda \mathbf{y}$, where $\mathbf{S}_\lambda$ has rows $\mathbf{b}'\left(x_i\right)$; as for splines λ can be chosen by cross-validation (but isn't on R).

- A (possibly) robust version of local polynomial regression (for $r = 0, 1, 2$) is incorporated in R, as the function `loess(...)`. Important options are

  1. `span` – related to λ; the default of .75 often gives too much smoothness.

  2. `family` – 'gaussian' for smoothly weighted (but not with a gaussian kernel, despite the name - see the `help` file) least squares fitting, 'symmetric' for fitting using a 'redescending M-estimation' procedure in place of least squares (more on this later).

[natheight=18.2056cm, natwidth=18.2056cm, height=15.5521cm, width=16.4571cm] C:/sw50/temp/graphics/mcycle2$_{f}ig$4$_{20.pdf}$

Loess fits. (a) Locally linear; “gaussian” family. (b) Locally quadratic; “gaussian” family. (c) Locally linear; “symmetric” family. (d) Locally quadratic; “symmetric” family.

14. Generalized additive modelling; Projection pursuit

- In many regression situations in which Y is to be modelled in terms of input variables $x_1, ..., x_p$, it may well be the case that the effects are additive BUT some of them are nonlinear in nature:

  $$E\left[Y|x_1, ..., x_p\right] = \alpha + f_1\left(x_1\right) + \cdots + f_p\left(x_p\right)$$

  for possibly nonlinear functions f_j. Here we look at a method to obtain nonparametric estimates of these functions. The objective is similar to that in spline fitting, in that we aim to minimize the penalized sum of squares

  $$\begin{aligned} &\sum_{i=1}^{n} \left\{y_i - \left(\alpha + f_1\left(x_{i1}\right) + \cdots + f_p\left(x_{ip}\right)\right)\right\}^2 \\ &+ \sum_{j=1}^{p} \lambda_j \int \left[f_j''\left(t\right)\right]^2 dt, \end{aligned}$$

  for chosen $\lambda_1, ..., \lambda_p \geq 0$.

- The parameter α is not identifiable (why not?). It becomes so if we impose the requirement $\sum_{i=1}^{n} f_j\left(x_{ij}\right) = 0$ for each j; then $\hat{\alpha} = \bar{y}$. From our previous work on splines it should not be surprising that the solution to the problem is now to fit a separate cubic spline to each f_j, with knots (i.e., nodes) at the unique values of the x_{ij}. The algorithm is as follows.

1. Initialize: $\hat{\alpha} = \bar{y}$, $\hat{f}_j \equiv 0$ for each j.

2. For $j = 1, ..., p$:

   (a) Fit a smoothing spline to f_j in the model

   $$y_i - \hat{\alpha} - \sum_{k \neq j} \hat{f}_k\left(x_{ik}\right) = f_j\left(x_{ij}\right) + \varepsilon_i.$$

   (b) Replace $\hat{f}_j\left(x_{ij}\right)$ by $\hat{f}_j\left(x_{ij}\right) - \frac{1}{n}\sum_{i=1}^{n} \hat{f}_j\left(x_{ij}\right)$.

3. Iterate to convergence.

This is carried out by the `gam(...)` function on R. It requires that one first load the package ‘gam’. This example uses the ‘rock’ data – permeability Y is to be modelled from three other variables: area, perimeter and shape (`help(rock)` for details). Since the range of Y is huge it is advised to use log($perm$).

```
# First do a linear fit:
rock.lm = lm(log(perm) ~area + peri + shape)
# gam fit:
rock.gam1 = gam(log(perm) ~s(area) + s(peri)
      + s(shape))
# Omitting the s() will result in linear
     terms being fitted.
par(mfrow=c(2,2))
plot(rock.gam1, se=T)
# The option 'se=T') results in +/- 2*std.err.
# confidence bands being plotted
```

[natheight=17.6279cm, natwidth=17.6718cm, height=16.7229cm, width=16.3275cm]
C:/sw50/temp/graphics/rock$_{fig1_{21.pdf}}$
Spline fits to each variable.

```
# Compare the two fits:
anova(rock.lm, rock.gam)
Analysis of Variance Table
Model 1: log(perm) ~area + peri + shape
Model 2: log(perm) ~s(area) + s(peri) + s(shape)
   Res.Df    RSS      Df   Sum of Sq   F    Pr(>F)
1 44.0000 31.949
2 34.9997 26.059  9.0003     5.890  0.8789  0.5528
```

Here $F = \frac{SS_1 - SS_2}{\Delta df} / \hat{\sigma}_\varepsilon^2$ is an approximate test statistic to test for *H*: Model 1 vs. *K*: Model 2 . It seems that $\hat{\sigma}_\varepsilon^2 = \max(MS_1, MS_2)$ is used. Thus

$$F = \frac{SS_1 - SS_2}{\Delta df} \Bigg/ \frac{SS_2}{df_2} = \frac{5.89}{9.003} \Bigg/ \frac{26.059}{34.997} = .8789;$$

this is not significant. From the plots only shape seems nonlinear. Re-fit:

```
rock.gam2 = gam(log(perm) ~area + peri + s(shape))
anova(rock.gam2, rock.gam1)
```

```
Analysis of Deviance Table

Model 1: log(perm) ~area + peri + s(shape)
Model 2: log(perm) ~s(area) + s(peri) + s(shape)
 Resid. Df Resid. Dev Df Deviance P(>|Chi|)
1 41.0000 28.9992
2 34.9997 26.0589 6.0003   2.9403    0.6836
```

For two fits from the same family the default test statistic (`test = "F"` is an option in `anova(...)`) is

$$\chi^2 = \frac{SS_1 - SS_2}{\hat{\sigma}^2_{\varepsilon}} = \frac{2.9403}{\max\left(MS_1, MS_2\right)} = 3.949;$$

with $P\left(\chi^2_{6.0003} > 3.949\right) = .6836$.

```
anova(rock.lm, rock.gam2, rock.gam1)
## tests model 1 against model 2 against model 3
Model 1: log(perm) ~area + peri + shape
Model 2: log(perm) ~area + peri + s(shape)
Model 3: log(perm) ~s(area) + s(peri) + s(shape)
   Res.Df    RSS      Df Sum of Sq      F Pr(>F)
1 44.0000 31.949
2 41.0000 28.999  3.0000     2.950 1.3205 0.2833
3 34.9997 26.059  6.0003     2.940 0.6582 0.6835
```

- Projection pursuit: This can be viewed as an attempt to answer the following question. Suppose that the vector $\mathbf{x}$ of independent variables is (possibly) of high dimension p. Are there 'interesting' linear combinations $\boldsymbol{\alpha}'\mathbf{x}$ and possibly nonlinear transformations $f(\cdot)$ such that we might profitably model the data as

  $$y = \sum_{m=1}^{M} f_m\left(\boldsymbol{\alpha}_m^{'}\mathbf{x}\right) + \varepsilon$$

  for some small value of M?

- We assume that all $\|\boldsymbol{\alpha}\| = 1$ so that the terms are possibly of comparable scales. Even then there is a problem if the x's are not measured in the same units. We typically scale the x_j so that at least their magnitudes are comparable.

- Now $\boldsymbol{\alpha} \cdot \boldsymbol{\alpha}'\mathbf{x} = \boldsymbol{\alpha} \cdot \left(\boldsymbol{\alpha}'\boldsymbol{\alpha}\right)^{-1} \boldsymbol{\alpha}'\mathbf{x}$ looks like the predictions following a regression of data $\mathbf{x}$ on the single regressor $\boldsymbol{\alpha}$. In our previous terminology it ‘lies in col$\,(\boldsymbol{\alpha})$ $(= \boldsymbol{\alpha})$’, and the length (norm) of the vector of predictions is $|\boldsymbol{\alpha}'\mathbf{x}|$. We call $\boldsymbol{\alpha}'\mathbf{x}$ the ‘projection’ in the direction $\boldsymbol{\alpha}$; hence the name projection pursuit.

- The model is very general; as well as picking out individual x’s (e.g. $\boldsymbol{\alpha} = (1, 0, \cdots, 0)'$) we can model interactions and many other terms. For instance

$$\begin{aligned} x_1 x_2 &= \frac{1}{2}\left(\frac{x_1 + x_2}{\sqrt{2}}\right)^2 - \frac{1}{2}\left(\frac{x_1 - x_2}{\sqrt{2}}\right)^2 \\ &= f_1\left(\boldsymbol{\alpha}_1^{'}\mathbf{x}\right) + f_2\left(\boldsymbol{\alpha}_2^{'}\mathbf{x}\right) \text{ for} \\ \boldsymbol{\alpha}_1^{'} &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right),\ \boldsymbol{\alpha}_2^{'} = \left(\frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}}\right), \\ f_1(t) &= \frac{t^2}{2},\ f_2(t) = -\frac{t^2}{2}. \end{aligned}$$

- Algorithm. The aim is to minimize

$$\sum_{i=1}^{n}\left(y_i - \sum_{m=1}^{M} f_m\left(\boldsymbol{\alpha}_m^{'}\mathbf{x}_i\right)\right)^2.$$

First suppose $M = 1$, so that $\sum_{i=1}^{n}\left(y_i - f_1\left(\boldsymbol{\alpha}_1'\mathbf{x}_i\right)\right)^2$ is to be minimized. If $\boldsymbol{\alpha}_1'$ is given, then $f_1(\cdot)$ can be gotten as in gam fitting. On the other hand if f_1 is given, and we have a trial value $\boldsymbol{\alpha}_{(0)}$ of $\boldsymbol{\alpha}$, then it can be updated by Gauss-Newton + WLS: take a linear approximation

$$f_1\left(\boldsymbol{\alpha}'\mathbf{x}\right) \approx f_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}\right) + \dot{f}_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}\right)\left(\boldsymbol{\alpha} - \boldsymbol{\alpha}_{(0)}\right)'\mathbf{x},$$

then

$$\sum_{i=1}^{n}\left(y_i - f_1\left(\boldsymbol{\alpha}'\mathbf{x}_i\right)\right)^2 \approx$$

$$\sum_{i=1}^{n}\left\{\begin{array}{c}\left[\dot{f}_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)\right]^2 \cdot \\ \left(\frac{y_i - f_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)}{\dot{f}_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)} - \left(\boldsymbol{\alpha} - \boldsymbol{\alpha}_{(0)}\right)'\mathbf{x}_i\right)^2\end{array}\right\}$$

is minimized by WLS, leading to the updated

$$\boldsymbol{\alpha}_{(1)} = \boldsymbol{\alpha}_{(0)} + \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}\mathbf{z},$$

with

$$z_i = \frac{y_i - f_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)}{\dot{f}_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)}, \text{ and } w_i = \left[\dot{f}_1\left(\boldsymbol{\alpha}_{(0)}'\mathbf{x}_i\right)\right]^2.$$

For $M > 1$ this is applied as in gam fitting, using the residuals from all $M - 1$ other fits, at each stage. The value M can be chosen by stopping when the addition of another term does not improve the fit appreciably.

- Simulated example: The data are simulated as $Y = x_1 x_2 + \varepsilon$, where x_1 and x_2 are uniformly distributed over $[-1, 1]$ and the errors are $N(0, (.2)^2)$.

```
set.seed(14) # To duplicate example
x1 = runif(400,-1,1)
x2 = runif(400,-1,1)
eps = rnorm(400,0,.2)   # Y=X1*X2+error
y = x1*x2+eps
x = cbind(x1,x2)
out = ppr(x,y, nterms = 1, max.terms = 4)
```

```
Goodness of fit:
 1 terms  2 terms  3 terms  4 terms
29.64065 18.64340 17.61438 17.47704
```

Suggests using two terms only; try 3 out of curiosity:

```
out = ppr(x,y, nterms = 3, max.terms = 4)
summary(out)
Goodness of fit:
 3 terms  4 terms
17.61438 17.47704

Projection direction vectors:
   term 1     term 2     term 3
x1 -0.7358552  0.7235624  0.8193227
x2 -0.6771389 -0.6902590 -0.5733327

Coefficients of ridge terms:
    term 1     term 2     term 3
0.19907585 0.19127321 0.05809677
```

Interpretation (+ hindsight!):

$$Y \;\approx\; \bar{Y} + \left[\begin{array}{c} \beta_1 f_1\left(-\frac{x_1+x_2}{\sqrt{2}}\right) + \beta_2 f_2\left(\frac{x_1-x_2}{\sqrt{2}}\right) \\ +\beta_3 f_3\left(.8x_1 - .6x_2\right) \end{array}\right] \tag{14.1}$$

$$\text{with } \beta_1 = .20,\ \beta_2 = .19,\ \beta_3 = .06$$
$$\text{and } \sum_i f_j\left(\boldsymbol{\alpha}'\mathbf{x}_i\right) = 0.$$

From the plots below (obtained by '`plot(out)`') and '`plot(x1*x2, out$fitted+mean(y))`' we obtain

$$f_1(t) \approx 3t^2 - 1,\ f_2(t) \approx 1 - 3t^2,\ f_3(t) \approx ??.$$

Using only the first two predictors in (14.1) gives

$$\begin{aligned} Y - \bar{Y} \;&\approx\; \beta_1 f_1\left(-\frac{x_1+x_2}{\sqrt{2}}\right) + \beta_2 f_2\left(\frac{x_1-x_2}{\sqrt{2}}\right) \\ &=\; \beta_1\left[\frac{3x_1^2 + 6x_1x_2 + 3x_2^2 - 2}{2}\right] \\ &\quad -\beta_2\left[\frac{3x_1^2 - 6x_1x_2 + 3x_2^2 - 2}{2}\right] \\ &\approx\; 6\bar{\beta}x_1x_2. \end{aligned}$$

[natheight=18.2056cm, natwidth=18.2056cm, height=11.679
width=12.1825cm] C:/sw50/temp/graphics/ppreg$_{sim_{22.pdf}}$

- Back to the ‘rock’ data.

```
shape1 = 100*shape
area1 = area/400
peri1 = peri/100
c(mean(shape1), mean(area1), mean(peri1))
[1] 21.81104 17.96932 26.82212
rock.ppr = ppr2(log(perm) ~area1 + peri1 + shape1,
  nterms = 2, max.terms = 4)

Goodness of fit:
 2 terms  3 terms  4 terms
9.620610 4.914191 4.294387

Projection direction vectors:
       term 1      term 2
area1   0.83001827  0.89216618
peri1  -0.55639534 -0.34630254
shape1  0.03865091  0.29002423

Coefficients of ridge terms:
   term 1    term 2
```

```
1.5752785 0.5296323
# Suggests area1 and peri1 alone
       determine log(perm)
```

[natheight=18.2056cm, natwidth=18.2056cm, height=17.274 width=18.2759cm] C:/sw50/temp/graphics/rock$_{fig}3_{23.pdf}$
To my eye, and suggested by the 'term1' and 'term2' plots above, we do as well with a second order linear model in 'area' and 'peri' ('rock.lm2' in these plots):

```
Residuals:
     Min       1Q   Median       3Q      Max
-1.50091 -0.38646 -0.00466  0.48448  1.42574

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.617947   0.541608  10.373 3.72e-13
area1        0.521732   0.125076   4.171 0.000149
peri1       -0.360705   0.077052  -4.681 2.97e-05
I(area1^2)  -0.023384   0.005697  -4.105 0.000182
I(peri1^2)  -0.004253   0.001716  -2.479 0.017278
area1:peri1  0.021962   0.004552   4.825 1.87e-05
---

Residual standard error: 0.6953 on
  42 degrees of freedom
Multiple R-squared:  0.8401,
  Adjusted R-squared:  0.821
F-statistic: 44.12 on 5 and 42 DF,
  p-value: 1.183e-15
```

15. Lasso; $n << p$; Quantile Regression

- Ridge regression can be viewed as 'shrinkage' – large values of $\left\|\hat{\boldsymbol{\theta}}\right\|^2$ are penalized, so some $\hat{\theta}_j$ 'shrink towards zero'. A solution still exists if $n < p$, but the biases get too large.

- In data mining for instance, one often encounters situations of very high dimensionality – p much larger than n ($n << p$). In such a situation one wants to eliminate variables entirely (not just assign them a small coefficient). A currently popular method is the 'lasso' – solve

$$\hat{\boldsymbol{\theta}} = \text{arg min} \left\|\mathbf{y} - \mathbf{X}\theta\right\|^2 \text{ s.t. } \sum \left|\theta_j\right| \leq t,$$

for some s. Equivalently, for some λ,

$$\hat{\boldsymbol{\theta}} = \text{arg min} \left\|\mathbf{y} - \mathbf{X}\theta\right\|^2 + \lambda \sum \left|\theta_j\right|.$$

- Recall that ridge regression uses the penalty $\sum \theta_j^2$.

- For the lasso, as t becomes large enough (larger than $t_0 = \sum \left|\hat{\theta}_j^{LS}\right|$) one recovers least squares.

[natheight=23.03cm, natwidth=29.8038cm, height=10.3637cm, width=13.7531cm]
C:/sw50/temp/graphics/lasso1$_{24.jpg}$
Contours of $\|\mathbf{y} - \mathbf{X}\beta\|^2$ and constraint regions; lasso vs. ridge. For small t the lasso solutions tend to set many $\hat{\theta}_j = 0$, as desired.

- Computing: See R code on course website; uses the 'glmnet' package.

- In the example the data are simulated in such a way that only the first coefficient should be meaningful.

```
library("glmnet")
set.seed(1)
n=100
p=5
X = matrix(rnorm(n*p), ncol = p)
y = X[,1] + rnorm(n) # = X*e1 + eps
```

[natheight=16.4108cm, natwidth=16.4858cm, height=8.4658cm, width=8.9073cm] C:/sw50/temp/graphics/lasso1$_{25.pdf}$

When $p < n$, horizontal axis is $s = t/t_0$, where $t_0 = \sum \left|\hat{\theta}_j^{LS}\right|$. So $s = 1$ recovers the LS estimates.

```
out = glmnet(X, y, int = F)
> # Look at the output:
      Df    %Dev   Lambda
 [1,]  0 0.00000 0.795800
 [2,]  1 0.06908 0.725100
 [3,]  1 0.12640 0.660700
              ....
[39,]  4 0.41550 0.023200
[40,]  4 0.41570 0.021140
[41,]  4 0.41590 0.019260
             .....
[62,]  5 0.41700 0.002730
[63,]  5 0.41700 0.002488
```

```
out$beta  # the coefs at each of the 63 stages
V1 . 0.07795298 0.1489808 0.2136988
V2 . .          .         .
V3 . .          .         .
V4 . .          .         .
V5 . .          .         .

          .....
V1  0.88001531  0.88034246  0.88064055
V2 -0.01423639 -0.01450489 -0.01474953
V3 -0.02382875 -0.02407353 -0.02429657
V4 -0.09499016 -0.09522749 -0.09544373
V5 -0.05387525 -0.05409895 -0.05430277

fit = lsfit(X,y, int = F)
fit$coef
         X1          X2          X3
 0.88369785 -0.01725878 -0.02658416
          X4          X5
 -0.09766166 -0.05639331
```

Now take $p = 150$:

[natheight=17.6279cm, natwidth=17.6718cm, height=13.6718cm, width=14.2078cm] C:/sw50/temp/graphics/lasso2$_{2}6.pdf$ Now, in the upper plot, the horizontal axis is $\sum_{j=1}^{p'} \left|\hat{\theta}j\right|$, where p' is the number of coefficients fitted; and $\sum_{j=1}^{150} \left|\hat{\theta}j\right| = 9.22$. In the lower plot all 150 final estimates are plotted; 60 of them are 0.

Quantile regression

- The solution $q(\mathbf{x}) = \theta_\tau'\mathbf{x}$ to $P\left(Y_{|\mathbf{x}} \leq q(\mathbf{x})\right) = \tau$ is the τ-regression quantile:
$$q(\mathbf{x}) = G^{-1}_{Y_{|\mathbf{x}}}(\tau).$$
With additive errors, this is $\tau = G_\varepsilon(0)$. If $\tau = .5$ one obtains the median (conditional on $\mathbf{x}$); if $\varepsilon = Y - \theta'_{.5}\mathbf{x}$ is symmetrically distributed then $q(\mathbf{x}) = E[Y|\mathbf{x}]$.

- Determined by
$$\hat{\boldsymbol{\theta}}_\tau = \text{arg}\min_t \sum_{i=1}^n \rho_\tau\left(Y_i - \mathbf{x}_i't\right),$$
where $\rho_\tau(\cdot)$ is the ‘check’ function
$$\rho_\tau(r) = r(\tau - I(r < 0)).$$
Equivalently
$$\sum_{i=1}^n \psi_\tau\left(Y_i - \mathbf{x}_i'\theta\right)\mathbf{x}_i = \mathbf{0},$$
with $\psi_\tau(r) = \tau - I(r < 0)$.

- – What are these if the only parameter is an intercept and $\tau = .5$?

[natheight=9.6019cm, natwidth=9.6019cm, height=7.273cm, width=7.273cm]
C:/sw50/temp/graphics/check$_{27.pdf}$
Check function; $\tau = .95$.

- Use ‘quantreg’ on R. Documentation by R. Koenker on course website.

- Example: Engel (1857) data on the relationship between food expenditure and household income.

```
y = engel$foodexp
x = engel$income
fit1 <- rq(y ~x, tau = 0.5)
fit1
summary(fit1, se = "nid") # Uses the
           Normal approximation below
```

```
Coefficients:
            Value    Std. Error t value  Pr(>|t|)
(Intercept) 81.48225 19.25066    4.23270  0.00003
x            0.56018  0.02828   19.81032  0.00000

# Several values of tau can be handled:
plot(x, y, cex = 0.25, type = "n", xlab =
 "Household Income", ylab = "Food Expenditure")
points(x, y, cex = 0.5, col = "black")
abline(rq(y ~x, tau = 0.05), col = "blue")
abline(rq(y ~x, tau = 0.95), col = "red")
```

[natheight=16.6311cm, natwidth=16.7178cm,
height=9.3512cm, width=9.4004cm]
C:/sw50/temp/graphics/quantreg$_{28.pdf}$
Quantile regression output; Engel data.
Interpretation?

- **Inferences**. There is a Normal approximation: as $n \to \infty$,
$$\hat{\boldsymbol{\theta}}_\tau - \theta_\tau \sim AN\left(0, w^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right),$$
where $w^2 = \tau\left(1-\tau\right)/g_\varepsilon^2\left(0\right)$.

Part IV

Robust Regression Methods

16. The need for robustness; M-estimation

- Good reading material for these lectures on robust regression:

    1. Maronna, Martin & Yohai; Chapters 4, 5.

    2. Rousseeuw & Leroy; Chapters 2, 3, 6.

- Robustness deals with the behaviour of statistical methods under violations of the assumptions, and with the derivation of methods which work 'almost' as well when these assumptions are violated as when they hold.

- Under what assumptions is Least Squares an optimal estimation method? This is answered by the *Gauss-Markov Theorem*: Consider the linear model $\mathbf{Y} = \mathbf{X}\boldsymbol{\theta} + \boldsymbol{\varepsilon}$, with uncorrelated, equally varied errors ε and with $\mathbf{X}_{n \times p}$ having full column rank. Suppose that we seek to estimate a linear combination $\alpha = \mathbf{a}'\boldsymbol{\theta}$ and require a *linear, unbiased* estimate: $\hat{\alpha} = \mathbf{c}'\mathbf{Y}$, $E[\hat{\alpha}] = \alpha$. Then the minimum variance estimate in this class, i.e. the 'Best Linear Unbiased Estimate' (BLUE), is

$$\begin{aligned} \hat{\alpha}_{BLUE} &= \mathbf{a}'\hat{\boldsymbol{\theta}}_{OLS} \\ (&= \mathbf{a}'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Y}, \text{ so } \mathbf{c} = \mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{a}). \end{aligned}$$

Proof: We are to show that $\hat{\alpha}_{BLUE}$ is unbiased (this is immediate) and that no unbiased estimate $\mathbf{c}'\mathbf{Y}$ has a smaller variance. That $\mathbf{c}'\mathbf{Y}$ be unbiased entails (how?)

$$\mathbf{X}'\mathbf{c} = \mathbf{a}, \tag{16.1}$$

and so we must show that, for any $\mathbf{c}$ satisfying (1), we have

$$\begin{aligned} \text{var}\left[\hat{\alpha}_{BLUE}\right] &\leq \text{var}\left[\mathbf{c}'\mathbf{Y}\right], \text{ i.e.} \\ \mathbf{a}'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{a} &\leq \mathbf{c}'\mathbf{c}. \end{aligned} \tag{16.2}$$

But in the presence of (16.1), (16.2) becomes $\mathbf{c}'H\mathbf{c} \leq \mathbf{c}'\mathbf{c}$, i.e. $\|(\mathbf{I} - \mathbf{H})\,\mathbf{c}\|^2 \geq 0$. □

- Note that the Gauss-Markov Theorem makes no assumptions about the distribution of the errors. They can be non-normal, and OLS is still optimal if (i) these errors are uncorrelated and homoscedastic, and (ii) we insist on a linear estimate. To improve on OLS we should drop the requirement of unbiasedness (recall ridge estimation) and/or look among non-linear estimates.

- Large sample inferences impose a further requirement. We typically carry out inferences about $\mathbf{a}'\boldsymbol{\theta}$ by using the normal approximation

$$\mathbf{a}'\hat{\boldsymbol{\theta}}_{OLS} \overset{d}{\approx} N\left(\mathbf{a}'\boldsymbol{\theta}, \sigma_\varepsilon^2 \mathbf{a}'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{a}\right),$$

  valid asymptotically even for non-normal errors under a condition that states, roughly, that no observations can have too large an influence on the fit. More precisely, *in order that all LSEs $\mathbf{a}'\hat{\boldsymbol{\theta}}_{OLS}$ be asymptotically normal, it is necessary and sufficient that 'Huber's condition' hold:*

$$\max_i h_{ii} \to 0 \text{ as } n \to \infty.$$

Recall that we tend to be wary of observations with $h_{ii} > 2\bar{h} = 2p/n$.

- Observations which dominate the LS fit due to unusual x-values are 'leverage' points (and the h_{ii} are sometimes called leverage values). Observations with unusually large (in absolute value) y-values are 'outliers'. These can arise from measurement error, instrument failure, incompetent sampling, ... It should be fairly clear that no linear estimate can be very good in the presence of outlying Y-values – think about the (linear) sample average vs. the (non-linear) sample median.

- One way in which outlying Y-values are sometimes modelled is by assuming that the errors follow a 'gross errors' model:

$$\varepsilon \sim (1-\alpha)\,\Phi\left(\frac{\varepsilon}{\sigma}\right) + \alpha G\left(\varepsilon\right),$$

where $\Phi(\varepsilon)$ is the $N(0,1)$ d.f. (so $\Phi\left(\frac{\varepsilon}{\sigma}\right)$ is $N\left(0,\sigma^2\right)$) and $G(\varepsilon)$ is an *arbitrary* d.f. The interpretation is that, with probability $1-\alpha$, an observation is drawn from the (ideal) $N\left(0,\sigma^2\right)$ population. With small probability α it is drawn from a population about which we have no knowledge.

- Effect of outliers and highly influential values

[natheight=18.2056cm, natwidth=18.2056cm, height=7.3521cm, width=12.1825cm] C:/sw50/temp/graphics/influence$_{fig1_{29.pdf}}$ Simulated data: $Y = x + \varepsilon$ with, on the left, one additional observation which is both highly influential (extreme $x-$ value) and has an outlying $y-$ value. Plot on right is after removal of this point.

- Since LS = ML for Normal errors, in looking for robust alternatives we might start with ML estimation for other distributions. An **M**-estimate is a generalization of a **M**aximum Likelihood estimate.

- If $Y_i = \mathbf{x}_i^{\prime}\boldsymbol{\theta} + \varepsilon_i$, with

$$\varepsilon_i \sim F\left(\frac{\varepsilon}{\sigma}\right), \text{ density } \frac{1}{\sigma}f\left(\frac{\varepsilon}{\sigma}\right),$$

then Y_i has density $\frac{1}{\sigma}f\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right)$ and the log-likelihood is

$$l\left(\boldsymbol{\theta},\sigma\right) = -n\log\sigma + \sum_i \log f\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right).$$

The MLE is then the minimizer of

$$\frac{1}{n}\sum_i \rho\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right) + \log\sigma,$$

where $\rho(r) = -\log f(r)$; this leads to the likelihood equations

$$\begin{aligned}\frac{1}{n}\sum_i \psi\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right)\mathbf{x}_i &= \mathbf{0}, \\ \frac{1}{n}\sum_i \psi\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right)\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right) - 1 &= 0, \quad (16.3)\end{aligned}$$

with ‘score function’ $\psi = \rho' = -f'/f$. For Normal errors, $\rho(r) = r^2/2$ and $\psi(r) = r$. (Note that any constant multiple of ψ can be used here instead.)

- What about Laplace errors $f(r) = .5e^{-|r|}$?

- For given ρ or ψ, what is f? (Conditions on ρ: $\rho(r) \to \infty$ sufficiently quickly as $|r| \to \infty$.)

- An M-estimate of regression is a solution to

$$\begin{aligned}\frac{1}{n}\sum_i \rho\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right) &= \text{min, or of} \\ \frac{1}{n}\sum_i \psi\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right)\mathbf{x}_i &= \mathbf{0},\end{aligned}$$

where $r_i\left(\hat{\boldsymbol{\theta}}\right) = y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}$ is the residual, and $\hat{\sigma}$ is an estimate of scale, perhaps determined by (16.3).

17. Huber's ψ_c; Computing M-estimates

- The LS estimate ($\psi(r) = r$) allows large residuals to have a large influence on the fit, and is non-robust for this reason. The L1 estimate ($\psi(r) = sgn\,(r)$) gives all residuals the same influence; for this reason it is highly robust but not very efficient if the errors are Normal. A compromise is 'Huber's ψ_c':

$$\psi_c\,(r) = \begin{cases} r, & |r| \le c, \\ c \cdot sgn(r), & |r| \ge c. \end{cases}$$

  - If this is an MLE, then what is f? Solving $\psi_c = -f'/f$ results in

$$f\,(r) = \begin{cases} A\phi\,(r)\,, & |r| \le c, \\ A\phi\,(c)\, e^{-c(|r|-c)}, & |r| \ge c, \end{cases}$$

    with $A \in (0, 1)$ determined from $\int f\,(r)\,dr = 1$:

$$\frac{1}{A} = 2\Phi\,(c) - 1 + 2\frac{\phi(c)}{c} \overset{?}{>} 1.$$

- This is a member of a gross errors neighbourhood of the Normal:
$$f\left(r\right)=\left(1-\alpha\right)\phi\left(r\right)+\alpha g\left(r\right)$$
for
$$\begin{aligned} A &= 1-\alpha, \\ g\left(r\right) &= \begin{cases} \frac{1-\alpha}{\alpha}\left[\phi\left(c\right)e^{-c\left(\left|r\right|-c\right)}-\phi\left(r\right)\right], & \left|r\right|\geq c, \\ 0, & \left|r\right|\leq c. \end{cases} \end{aligned}$$
To establish this requires showing that, for any $\alpha\in\left(0,1\right)$ there exists $A\in\left(0,1\right)$ and $c>0$ satisfying these equations, and that $g\left(r\right)$ is a valid density.

  - In practice, one typically takes $c\in\left(1,2\right)$.

- Computing. Suppose first that scale σ is known, and so we wish only to compute $\hat{\boldsymbol{\theta}}$ by solving
$$\begin{aligned} \frac{1}{n}\sum_i\rho\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\sigma}\right) &= \text{min, or} \\ \frac{1}{n}\sum_i\psi\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\sigma}\right)\mathbf{x}_i &= \mathbf{0}. \qquad (17.1) \end{aligned}$$

Suppose as well that ψ is monotone, i.e. $\psi' \geq 0$. If ψ is *strictly* increasing then we are guaranteed a unique solution, since the function being minimized is convex:

$$\frac{\partial^2}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}}\frac{1}{n}\sum_i \rho\left(\frac{r_i(\boldsymbol{\theta})}{\sigma}\right) = \frac{1}{n\sigma^2}\sum_i \psi'\left(\frac{r_i(\boldsymbol{\theta})}{\sigma}\right)\mathbf{x}_i\mathbf{x}_i'$$

is positive definite:

$$\begin{aligned} &\mathbf{c}'\left[\sum_i \psi'\left(\frac{r_i(\boldsymbol{\theta})}{\sigma}\right)\mathbf{x}_i\mathbf{x}_i'\right]\mathbf{c} \\ = \ &\sum_i \psi'\left(\frac{r_i(\boldsymbol{\theta})}{\sigma}\right)\left(\mathbf{c}'\mathbf{x}_i\right)^2 > 0. \end{aligned}$$

For Huber's ψ we have only '≥ 0', and indeed $\boldsymbol{\theta}$ can be chosen so badly that all $|r_i(\boldsymbol{\theta})| > c\sigma$ and so all $\psi'(r_i(\boldsymbol{\theta})/\sigma)$ are $= 0$. But the objective function is still convex in a neighbourhood of a solution $\hat{\boldsymbol{\theta}}$ for which most of the residuals satisfy $\left|r_i\left(\hat{\boldsymbol{\theta}}\right)\right| \leq c\sigma$.

- We will always assume that ψ is an odd function (and so $\psi(0) = 0$ if ψ is continuous), and $\psi(r) \geq 0$ for positive r. Introduce 'weights' $w(x) = \psi(x)/x$ ($= \psi'(0)$ at $x = 0$); thus $w(x)$ is even and everywhere non-negative. Then with $w_i = w\left(r_i\left(\hat{\boldsymbol{\theta}}\right)/\sigma\right)$, (17.1) can be written
$$\frac{1}{n}\sum_i w_i\left(y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}\right)\mathbf{x}_i = \mathbf{0},$$
with 'solution'
$$\hat{\boldsymbol{\theta}} = \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}.$$
This is only a 'solution' because the weights depend on $\hat{\boldsymbol{\theta}}$. But we can iterate:

1. Start with $\boldsymbol{\theta}_{(0)}$; compute residuals $y_i - \mathbf{x}_i'\boldsymbol{\theta}_{(0)}$ and weights $w_{i,(0)} = w\left(r_i\left(\boldsymbol{\theta}_{(0)}\right)/\sigma\right)$.

2. Do a WLS regression of $\mathbf{y}$ on $\mathbf{X}$ with weights $w_{i,(0)}$ to obtain $\boldsymbol{\theta}_{(1)}$.

3. Iterate to convergence. (Why is the converged value a solution?)

This is called ‘Iteratively Reweighted Least Squares’ (IRLS).

- When scale is to be estimated as well, we replace σ by $\hat{\sigma}$ in these expressions, and update it along with $\boldsymbol{\theta}$: after $\boldsymbol{\theta}_{(k)}$ and $r_i\left(\boldsymbol{\theta}_{(k)}\right)$ have been obtained, update $\hat{\sigma}$ to $\sigma_{(k+1)}$. One proposal is to solve

  $$\frac{1}{n-p}\sum_i \psi^2\left(\frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right) = E_{\Phi}\left[\psi^2\left(\varepsilon\right)\right] \overset{def}{=} \delta$$

  (in analogy with LS) through:

  $$\sigma^2_{(k+1)} = \frac{1}{\delta\left(n-p\right)}\sum_i w^2_{i(k)} r_i^2\left(\boldsymbol{\theta}_{(k)}\right).$$

  For Huber’s ψ_c,

  $$\delta = 1 - 2c\phi(c) + 2\left(c^2-1\right)\Phi\left(-c\right).$$

- A commonly used alternative is the median absolute deviation (MAD):

$$\sigma_{(k+1)} = \frac{med\left\{\left|r_i\left(\boldsymbol{\theta}_{(k)}\right)\right|\right\}}{.6745}.$$

The denominator ($= \Phi^{-1}(.75)$) is such that, at the Normal distribution, the estimate is consistent (tends in probability to σ as $n \to \infty$). Outline:

$$med\{sample\} \overset{pr}{\to} med\{population\} = F^{-1}(.5),$$

where

$$F(t) = P_\Phi\left(|Z| \le t\right) = 2\Phi(t) - 1.$$

Thus for N(0,1) errors, $F^{-1}(.5) = \Phi^{-1}(.75)$ and

$$med\left\{\left|\frac{r_i}{\sigma}\right|\right\} \overset{pr}{\to} med\{|Z|\} = \Phi^{-1}(.75).$$

- R-code (on course website):

```
# Set ’c’:
c = 1

# Define Huber’s psi function:
psi = function(r) pmax(-c, pmin(r,c))

# Weights:
w = function(r) pmin(1, c/abs(r))

# Delta:
delta = 1-2*c*dnorm(c)+2*(c^2-1)*pnorm(-c)

# Arrange to store the output:
out = matrix(ncol = p+2)
dimnames(out) = ...

# Start with an L1-estimate
library(quantreg)
init.fit = rq(y ~x)
theta = init.fit$coef
```

```
r = init.fit$resid
sigma = mad(r, center = 0) #Initial scale
std.res = r/sigma
weights = w(std.res)
norm = sqrt(sum((t(X)%*%psi(std.res))^2))
 #Euclidean norm of t(X)%*%psi(std.res); = 0?
out[1,] = round(c(theta, sigma, norm),5)

while (norm > .001) {
fit = lsfit(x, y, wt = weights)
theta = fit$coef
r = fit$resid
sigma = sqrt(sum((weights*r)^2)/(delta*(n-p)))
std.res = r/sigma
norm = sqrt(sum((t(X)%*%psi(std.res))^2))
out = rbind(out, round(c(theta, sigma, norm),5))
weights = w(std.res)
}
```

Here is (some of) the output with the ‘stackloss’ data:

```
> print(out[,-c(1:2)])
      water.temp acid.conc   sigma      norm
 [1,]    0.57391  -0.06087 1.75334 118.21923
 [2,]    0.71207  -0.10581 1.92218  14.68070
 [3,]    0.75289  -0.10812 2.10807  16.76062
 ...
[48,]    0.87375  -0.12075 2.78565   0.00101
[49,]    0.87375  -0.12075 2.78565   0.00081
> # Compare with Least Squares:
> print(lsfit(x,y)$coef[3:4])
Water.Temp Acid.Conc.
 1.2952861 -0.1521225
> print(round(weights,2))
 [1] 0.82 1.00 0.63 0.41 1.00 1.00 1.00 1.00 1.00
 [10]1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00
 [19] 1.00 1.00 0.31
```

[natheight=18.2693cm, natwidth=18.2693cm, height=16.255 width=17.7421cm] C:/sw50/temp/graphics/robust$_{s}tackloss_{3}$

18. Asymptotics; Inferences; Pseudovalues

Making inferences after M-estimation requires an approximate distribution of $\hat{\boldsymbol{\theta}}$. Here is an outline of the derivation. Assume scale is known, so that $\hat{\boldsymbol{\theta}}$ is a solution to $\mathbf{G}\left(\hat{\boldsymbol{\theta}}\right)=\mathbf{0}$, where

$$\mathbf{G}\left(\boldsymbol{\theta}\right)=\frac{1}{n}\sum_{i}\psi\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}}{\sigma}\right)\mathbf{x}_i.$$

We expand around the true $\boldsymbol{\theta}$ – call this $\boldsymbol{\theta}_0$. It is *defined* by

$$E\left[\mathbf{G}\left(\boldsymbol{\theta}_0\right)\right]=\mathbf{0};$$

this is guaranteed for an ordinary M-estimate with an odd ψ-function and symmetrically distributed errors $\varepsilon_i=y_i-\mathbf{x}_i'\boldsymbol{\theta}_0$, since then

$$E\left[\psi\left(\frac{\varepsilon}{\sigma}\right)\right]=0.$$

The expansion is

$$\mathbf{0}=\mathbf{G}\left(\hat{\boldsymbol{\theta}}\right)=\mathbf{G}\left(\boldsymbol{\theta}_0\right)+\dot{\mathbf{G}}\left(\boldsymbol{\theta}_0\right)\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}_0\right)+R_n$$

for a remainder R_n; then

$$\begin{aligned}\sqrt{n}\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}_0\right) &= \left[-\dot{\mathbf{G}}\left(\boldsymbol{\theta}_0\right)\right]^{-1}\sqrt{n}\mathbf{G}\left(\boldsymbol{\theta}_0\right)\\ &\quad +\left[-\dot{\mathbf{G}}\left(\boldsymbol{\theta}_0\right)\right]^{-1}\sqrt{n}R_n.\end{aligned}$$

It can be shown that $\sqrt{n}R_n \overset{pr}{\to} 0$, so that the asymptotic distribution is the same as that of

$$\begin{aligned}&\left[-\dot{\mathbf{G}}\left(\boldsymbol{\theta}_0\right)\right]^{-1}\sqrt{n}\mathbf{G}\left(\boldsymbol{\theta}_0\right)\\ &= \left[\frac{1}{n\sigma}\sum_i \psi'\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}_0}{\sigma}\right)\mathbf{x}_i\mathbf{x}_i'\right]^{-1}\\ &\quad \cdot\frac{1}{\sqrt{n}}\sum_i \psi\left(\frac{y_i-\mathbf{x}_i'\boldsymbol{\theta}_0}{\sigma}\right)\mathbf{x}_i\\ &= \left[\frac{1}{n\sigma}\sum_i \psi'\left(\frac{\varepsilon_i}{\sigma}\right)\mathbf{x}_i\mathbf{x}_i'\right]^{-1}\cdot\frac{1}{\sqrt{n}}\sum_i \psi\left(\frac{\varepsilon_i}{\sigma}\right)\mathbf{x}_i.\end{aligned}$$

Recall the WLLN and the CLT (see Appendix). By these, the first term is asymptotically equal to the inverse of

$$\frac{1}{\sigma}E\left[\psi'\left(\frac{\varepsilon}{\sigma}\right)\right]\left[\frac{1}{n}\sum_i \mathbf{x}_i\mathbf{x}_i'\right],$$

and the second is asymptotically normally distributed, with mean zero (why?) and asymptotic covariance $E\left[\psi^2\left(\frac{\varepsilon}{\sigma}\right)\right]\left[\frac{1}{n}\sum_i \mathbf{x}_i\mathbf{x}_i'\right]$. It follows that

$$\sqrt{n}\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}_0\right)\overset{d}{\approx}N\left(\mathbf{0},\sigma^2\frac{E\left[\psi^2\left(\frac{\varepsilon}{\sigma}\right)\right]}{\left(E\left[\psi'\left(\frac{\varepsilon}{\sigma}\right)\right]\right)^2}\left[\frac{1}{n}\sum_i \mathbf{x}_i\mathbf{x}_i'\right]^{-1}\right).$$

When scale is estimated as well, we replace σ by $\hat{\sigma}$ in order to apply the approximation, which we can also write as

$$\hat{\boldsymbol{\theta}}\overset{d}{\approx}N\left(\boldsymbol{\theta}_0,V\left(\psi,F\right)\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right)$$

where $\varepsilon\sim F$ and

$$V\left(\psi,F\right)=\sigma^2\frac{E_F\left[\psi^2\left(\frac{\varepsilon}{\sigma}\right)\right]}{\left(E_F\left[\psi'\left(\frac{\varepsilon}{\sigma}\right)\right]\right)^2}. \qquad (18.1)$$

We estimate $V\left(\psi,F\right)$ by

$$v_\psi=\hat{\sigma}^2\frac{\frac{1}{n-p}\sum_i\psi^2\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right)}{\left[\frac{1}{n}\sum_i\psi'\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right)\right]^2}.$$

- You should check that these approximations are *exact* if LS is used and the errors are Normal.

- Inferences can be made in much the same way as when least squares estimates are used, after making appropriate modifications for the revised covariance structure of $\hat{\boldsymbol{\theta}}$. For instance, tests and confidence intervals on $\mathbf{a}'\boldsymbol{\theta}$ use the approximation

$$\mathbf{a}'\hat{\boldsymbol{\theta}} \overset{d}{\approx} N\left(\mathbf{a}'\boldsymbol{\theta}, V(\psi, F)\,\mathbf{a}'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{a}\right),$$

with

$$\frac{\mathbf{a}'\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)}{s_{\psi}\sqrt{\mathbf{a}'\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{a}}} \overset{d}{\approx} t_{n-p}$$

for $s_{\psi} \overset{def}{=} \sqrt{v_{\psi}}$.

- Note that the only change here (and in F-tests, etc.) is that the LS-based S is replaced by s_{ψ}.

- In LS regression, the t-ratios and p-values appear on the printout, which is very convenient. Is there a way to have the printout reflect these values after a robust regression? An easy way to accomplish this is to do a final least squares regression with the y_i replaced by 'pseudovalues'

  $$\begin{aligned} \tilde{y}_i &= \mathbf{x}_i'\hat{\boldsymbol{\theta}} + \frac{\hat{\sigma}}{a}\psi\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right), \text{ where} \\ a &= \frac{1}{n}\sum_i \psi'\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right). \end{aligned}$$

  The output will produce the LS-estimates

  $$\begin{aligned} \hat{\boldsymbol{\theta}}_{LS} &= \left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\tilde{\mathbf{y}} \\ &= \hat{\boldsymbol{\theta}} + \left(\mathbf{X}'\mathbf{X}\right)^{-1}\frac{\hat{\sigma}}{a}\sum \mathbf{x}_i\psi\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right) \\ &= \hat{\boldsymbol{\theta}}. \end{aligned}$$

  The inferences reported on the printout will be based on an estimated covariance matrix $S^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}$, with

$$\begin{aligned} S^2 &= \frac{\sum_i \left(\tilde{y}_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}\right)^2}{n-p} \\ &= \frac{\left(\frac{\hat{\sigma}}{a}\right)^2 \sum_i \psi^2\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right)}{n-p} \\ &= v_\psi. \end{aligned}$$

(See the Street, Carroll & Ruppert paper on the course website, for more on computing ordinary M-estimates.)

```
# Compute pseudovalues
psiprime = function(r) (abs(r)<=c)
a = mean(psiprime(std.res))
y.tilde = X%*%theta + (sigma/a)*psi(std.res)
pseudofit = lsfit(x,y.tilde)
ls.print(pseudofit)

Residual Standard Error=2.472
R-Square=0.9468
F-statistic (df=3, 17)=100.9487
p-value=0

           Estimate Std.Err t-value Pr(>|t|)
Intercept  -40.6590  9.0668 -4.4844   0.0003
Air.Flow     0.8302  0.1028  8.0772   0.0000
Water.Temp   0.8738  0.2805  3.1150   0.0063
Acid.Conc.  -0.1207  0.1191 -1.0137   0.3250
```

- To now we have implicitly treated the regressors $\mathbf{x}_i$ as fixed, i.e. non-random. In practice they are often observed values of random variables. Two possibilities arise. For simplicity take a straight line model

$$Y = \theta_0 + \theta_1 x + \varepsilon. \tag{18.2}$$

1. Random regressors – here x is assumed to be the observed value of a r.v. X, whose distribution is independent of that of ε, and does not depend on θ_0, θ_1 or σ_ε^2. Then if the *conditional* distribution of Y, given $X = x$, is normal (and homoscedastic, etc.), the usual Least Squares analysis is valid (conditionally):

$$\begin{aligned} E\left[\hat{\boldsymbol{\theta}}|\mathbf{X}\right] &= \boldsymbol{\theta}, \\ \text{cov}\left[\hat{\boldsymbol{\theta}}|\mathbf{X}\right] &= \sigma_\varepsilon^2\left(\mathbf{X}'\mathbf{X}\right)^{-1}. \end{aligned}$$

   We will take an analogous approach – in the model $Y_i = \mathbf{x}_i'\boldsymbol{\theta} + \varepsilon_i$ we assume that $\mathbf{x}_i$ and ε_i are independently (but perhaps not Normally) distributed. In the same way that outlying Y values receive reduced weights in a robust regression, we might want to bound the influence of the $\mathbf{x}_i$ (recall that we flag as highly influential those $\mathbf{x}_i$ with $h_{ii} > 2\bar{h}$).

2. Measurement errors, or ‘errors in variables’ models – here it is assumed that there is a ‘true’ value x of X, and that, rather than (18.2), one observes

$$\begin{aligned} Y &= \theta_0 + \theta_1 X + \varepsilon, \text{ with} \\ X &= x + \delta \end{aligned}$$

for a random error δ (typically assumed independent of ε). Then bias is introduced; this and other aspects are discussed in Draper & Smith (§3.4).

19. Breakdown and Influence

- The case of random regressors raises new robustness issues, since now there may be highly influential values of the $\mathbf{x}_i$. Example: the ‘mineral’ data set in Maronna, Martin & Yohai gives values of zinc vs. copper in 53 rock samples from Western Australia. One observation (#15) is a clear outlier. Both the LS line and a robust fit using Huber's $\psi_{1.5}$ are strongly influenced by this point. Also shown is the LS fit after removing this point.

[natheight=18.2056cm, natwidth=18.2056cm, height=16.764
width=18.2759cm] C:/sw50/temp/graphics/mineral1$_{31.pdf}$

- Two methods have been developed to assess the robustness of a regression fit. The first is the 'breakdown point' (BP). Roughly speaking, this is the largest fraction of data values which can be corrupted (made arbitrarily bad) with the estimates remaining bounded. Formally, for a data set $Z = \{\mathbf{x}_i, y_i\}_{i=1}^n$, let Z_m denote any data set with at least $n - m$ elements in common with Z (so at most m can be corrupted). Define

$$m^* = \max\left\{m \,|\, \hat{\boldsymbol{\theta}}\left(Z_m\right) \text{ is bounded for all } Z_m\right\}.$$

Then $\varepsilon^* = m^*/n$ is called the 'finite-sample breakdown point', and $\lim_{n\to\infty} \varepsilon^*$ is the breakdown point.

- Example: In a location model $y_i = \mu + \varepsilon_i$ the sample average $\bar{y}$ has $BP = 0$. (*Proof*: Let $y_1 \to \infty$, then $\bar{y} \to \infty$; thus $\varepsilon^* \leq 1/n \to 0$.)

- Example: In the same model the sample median $\tilde{y}$ has $BP = .5$. (*Proof*: Suppose $n = 2m$ is even and order the observations $y_{(1)} \leq \cdots \leq y_{(n)}$. Then $\tilde{y}$ is between $y_{(m)}$ and $y_{(m+1)}$. Clearly, the worst that can happen is that the largest observations are sent to $-\infty$ (or the smallest ...). In the first case the median does not drop below $y_{(1)}$ if this is done to groups of size $m-1$, but can be unbounded if groups of size m are altered in this way. In the second ... Thus $\varepsilon^* = (m-1)/n \to .5$.)

- In the regression model $Y_i = \mathbf{x}_i'\boldsymbol{\theta} + \varepsilon_i$ with random $\mathbf{x}_i$, an M-estimate of regression with *monotone* ψ has $BP = 0$.

  Proof: For simplicity, suppose $\sigma = 1$ is known. We show that it is possible to alter $(\mathbf{x}_1, y_1)$ in such a way that $\left\|\hat{\boldsymbol{\theta}}\right\| \to \infty$. Note that

  $$\psi\left(y_1 - \mathbf{x}_1'\hat{\boldsymbol{\theta}}\right)\mathbf{x}_1 + \sum_{i=2}^{n} \psi\left(y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}\right)\mathbf{x}_i = \mathbf{0}. \tag{19.1}$$

  Let y_1 and $\|\mathbf{x}_1\|$ both $\to \infty$ in such a way that $y_1/\|\mathbf{x}_1\| \to \infty$. Leave $\{y_i, \mathbf{x}_i\}_{i=2}^{n}$ fixed. Then

  $$y_1 - \mathbf{x}_1'\hat{\boldsymbol{\theta}} \geq y_1 - \|\mathbf{x}_1\|\left\|\hat{\boldsymbol{\theta}}\right\| = \|\mathbf{x}_1\|\left(\frac{y_1}{\|\mathbf{x}_1\|} - \left\|\hat{\boldsymbol{\theta}}\right\|\right).$$

  If $\left\|\hat{\boldsymbol{\theta}}\right\| \nrightarrow \infty$ then $y_1 - \mathbf{x}_1'\hat{\boldsymbol{\theta}} \to \infty$, and so $\psi\left(y_1 - \mathbf{x}_1'\hat{\boldsymbol{\theta}}\right) \to \psi(\infty) > 0$. Thus the norm of the first term in (19.1) $\to \infty$ and hence some of $\left\{\psi\left(y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}\right)\right\}_{i=2}^{n}$ must be unbounded. This is impossible if ψ is bounded, and if ψ is unbounded it can only happen if $\left\|\hat{\boldsymbol{\theta}}\right\| \to \infty$.

- In view of this last result, it is important to find robust estimates of regression with positive BPs (when contaminated regressors are a possibility). In fact $BP = .5$ is attainable. We will look at two possibilities – (i) expand the class of M-estimates to allow for the effect of influential $\mathbf{x}$'s to be bounded, or (ii) drop the requirement of a monotone ψ. The first of these leads to 'Bounded Influence' or 'Generalized' M-estimation, the second to 'MM-estimation'. These will each be discussed in the next few lectures.

- The second method developed to assess the robustness of a technique involves measuring the (asymptotic) influence of a data point on a statistic. Again there is a finite sample version and a limiting version. The first is the 'sensitivity curve' – let $T(\mathbf{z}_1, ..., \mathbf{z}_n)$ be a statistic computed from data $\{\mathbf{z}_1, ..., \mathbf{z}_n\}$ and consider

$$SC(\mathbf{z}) = T(\mathbf{z}_1, ..., \mathbf{z}_n, \mathbf{z}) - T(\mathbf{z}_1, ..., \mathbf{z}_n).$$

This measures the effect of adding one arbitrary observation to the finite sample.

Example: Let $T\left(y_1, ..., y_n\right) = \bar{y}$. Then $SC\left(y\right) = \frac{y-\bar{y}}{n+1}$, with

$$n \cdot SC\left(y\right) \overset{pr}{\rightarrow} y - \mu.$$

The influence of y on $\bar{y}$ is proportional to $y-\mu$, i.e. outliers have more influence!

- To get a limiting version of SC one might multiply by n and take a limit, as above. To define it more formally, we first need the 'empirical distribution function' (e.d.f.) $\hat{F}_n$ of a sample $\{\mathbf{z}_1, ..., \mathbf{z}_n\}$; this is the d.f. with

$$P_{\hat{F}_n}\left(\mathbf{z} = \mathbf{z}_i\right) = \frac{1}{n},\ i = 1, ..., n.$$

All of the common statistics can be defined as 'functionals' $h(\hat{F}_n)$ of the e.d.f. For instance

$$\bar{y} = \sum y_i P_{\hat{F}_n}\left(y = y_i\right) = E_{\hat{F}_n}\left[Y\right].$$

- Corresponding to a statistic $h(\hat{F}_n)$, suppose that F is the population d.f. Assume that $F \in \mathbb{F}$, a convex class of d.f.s. Consider

  $$\begin{aligned} \dot{h}(F_0; F_1) &= \lim_{t \to 0} \frac{h\left((1-t)F_0 + tF_1\right) - h\left(F_0\right)}{t} \\ &= \frac{d}{dt} h\left(F_t\right)_{|t=0}, \text{ where we define} \\ F_t &= (1-t)F_0 + tF_1. \end{aligned}$$

  When $F_1 = \delta_{\mathbf{z}}$ (point mass at $\mathbf{z}$) this represents the limiting, normalized influence of a new observation, with value $\mathbf{z}$, on the statistic $h\left(F_0\right)$. We call

  $$\dot{h}(F_0; \delta_{\mathbf{z}}) = IF(\mathbf{z}) \text{ (or } IF(\mathbf{z}; h, F_0))$$

  the *Influence Function*. It can be used as a measure of the robustness of a procedure against outliers (ideally we would like it to be bounded). We will see in the next class that it can also be used to give a quick asymptotic normality proof:

  $$\sqrt{n}\left(h(\hat{F}_n) - h\left(F_0\right)\right) \overset{d}{\approx} N\left(0, \text{var}_{F_0}\left[IF(\mathbf{Z}; h, F_0)\right]\right).$$

  .

Example: If $h(F) = E_F[Y]$, then

$$\begin{aligned} h\left(\hat{F}_n\right) &= \bar{y}, \\ h\left(F_t\right) &= h\left((1-t)F_0 + tF_1\right) \\ &= (1-t)h(F_0) + th(F_1), \\ \dot{h}(F_0; F_1) &= \left(h(F_1) - h(F_0)\right), \end{aligned}$$

and so

$$IF(y) = \dot{h}(F_0; \delta_y) = y - E_{F_0}[Y].$$

The IF is unbounded; this is evidence of the lack of robustness of the sample average. A single arbitrarily large outlier can push $\bar{y}$ beyond all bounds.

Note that $IF(y) = \lim n \cdot SC\,(y)$; also that

$$E_{F_0}[IF(Y)] = 0$$

in this example. This turns out to be true very generally; hence

$$\text{var}_{F_0}\left[IF(\mathbf{Z}; h, F_0)\right] = E_{F_0}\left[IF^2(\mathbf{Z}; h, F_0)\right].$$

20. Generalized M-estimation; High breakdown estimates

- **IF of an M-estimate**. Let $\hat{\boldsymbol{\theta}}$ be an M-estimate defined by

$$\frac{1}{n}\sum \psi\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{\sigma}}\right)\mathbf{x}_i = \mathbf{0}.$$

More generally, for a sample $\{\mathbf{z}_i = (\mathbf{x}_i, y_i)\}_{i=1}^n$ we can write

$$\boldsymbol{\Psi}\left(\mathbf{z}_i; \hat{\boldsymbol{\theta}}\right) = \psi\left(\frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\sigma}\right)\mathbf{x}_i$$

(assume for simplicity that σ is known) and define $\hat{\boldsymbol{\theta}}$ as a solution to

$$\frac{1}{n}\sum \boldsymbol{\Psi}\left(\mathbf{z}_i; \boldsymbol{\theta}\right) = \mathbf{0}.$$

This defines $\hat{\boldsymbol{\theta}}$ implicitly as a functional $h\left(\hat{F}_n\right)$ of the e.d.f. of $\{\mathbf{z}_i\}_{i=1}^n$:

$$E_{\hat{F}_n}\left[\boldsymbol{\Psi}\left(\mathbf{Z}; h\left(\hat{F}_n\right)\right)\right] = \mathbf{0}.$$

If F_0 is the distribution function of the $\mathbf{z}_i$ then the parameter $\boldsymbol{\theta}$ being estimated is defined by

$$E_{F_0}\left[\Psi\left(\mathbf{Z}; h\left(F_0\right)\right)\right] = \mathbf{0}. \qquad (20.1)$$

To calculate the IF, replace $\hat{F}_n$ by F_t:

$$E_{F_t}\left[\Psi\left(\mathbf{Z}; h\left(F_t\right)\right)\right] = \mathbf{0}.$$

Differentiate (and use (20.1)):

$$\begin{aligned} \mathbf{0} &= \frac{d}{dt} E_{F_t}\left[\Psi\left(\mathbf{Z}; h\left(F_t\right)\right)\right]_{|t=0} \\ &= E_{F_1}\left[\Psi\left(\mathbf{Z}; h\left(F_0\right)\right)\right] + E_{F_0}\left[\dot{\Psi}\left(\mathbf{Z}; h\left(F_0\right)\right)\right] \dot{h}(F_0; F_1). \end{aligned}$$

Thus

$$\dot{h}(F_0; F_1) = \left\{E_{F_0}\left[-\frac{\partial}{\partial \boldsymbol{\theta}}\Psi\left(\mathbf{Z}; \boldsymbol{\theta}\right)\right]\right\}^{-1} E_{F_1}\left[\Psi\left(\mathbf{Z}; \boldsymbol{\theta}\right)\right]$$

with

$$IF(\mathbf{z}) = \dot{h}(F_0; \delta_{\mathbf{z}}) = \left\{E_{F_0}\left[-\frac{\partial}{\partial \boldsymbol{\theta}}\Psi\left(\mathbf{Z}; \boldsymbol{\theta}\right)\right]\right\}^{-1} \Psi\left(\mathbf{z}; \boldsymbol{\theta}\right).$$

Note that $E_{F_0}[IF(\mathbf{Z})] = \mathbf{0}$.

- **Asymptotic normality**. By Taylor's Theorem, expanding $h\left(F_t\right)$ around $t=0$ gives

  $$\begin{aligned} h\left(F_t\right) &= h\left(F_0\right)+\dot{h}(F_0;F_1)t+..., \text{ whence} \\ h\left(F_1\right) &= h\left(F_0\right)+\dot{h}(F_0;F_1)+\text{Remainder.} \end{aligned}$$

  Typically (but <u>this has to be checked</u>)

  $$\dot{h}(F_0;F_1)=E_{F_1}\left[\Omega(\mathbf{Z})\right]$$

  for some vector $\Omega(\mathbf{z})$. With $F_1=\delta_{\mathbf{z}}$ we obtain $IF(\mathbf{z})=\Omega(\mathbf{z})$. Then with $F_1=F_0$ we obtain

  $$E_{F_0}[IF(\mathbf{Z})]=\mathbf{0}.$$

  Thus

  $$h\left(F_1\right)=h\left(F_0\right)+E_{F_1}\left[IF(\mathbf{Z})\right]+\text{Remainder}$$

  and then, with $F_1=\hat{F}_n$, we have (a "Mean Value Theorem")

  $$\sqrt{n}\left(h\left(\hat{F}_n\right)-h\left(F_0\right)\right)=\frac{1}{\sqrt{n}}\sum_{i=1}^{n}IF\left(\mathbf{Z}_i\right)+\sqrt{n}R_n,$$

  where the $IF\left(\mathbf{Z}_i\right)$ are i.i.d. r.vectors with mean $\mathbf{0}$ and variance

  $$\mathbf{\Sigma}\left(F_0\right)=E_{F_0}\left[IF\left(\mathbf{Z}\right)\cdot IF'\left(\mathbf{Z}\right)\right].$$

By the CLT, as long as $\sqrt{n}R_n \overset{pr}{\to} 0$ (and typically it does) we have

$$\sqrt{n}\left(h\left(\hat{F}_n\right) - h\left(F_0\right)\right) \overset{L}{\to} N\left(\mathbf{0}, \boldsymbol{\Sigma}\left(F_0\right)\right).$$

- Applied to an M-estimate, this gives

$$\sqrt{n}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \overset{L}{\to} N\left(\mathbf{0}, \mathbf{M}^{-1}\mathbf{Q}\mathbf{M}^{-1}\right)$$

with

$$\begin{aligned} \mathbf{M} &= E_{F_0}\left[-\frac{\partial}{\partial \boldsymbol{\theta}}\boldsymbol{\Psi}\left(\mathbf{Z};\boldsymbol{\theta}\right)\right], \\ \mathbf{Q} &= E_{F_0}\left[\boldsymbol{\Psi}\left(\mathbf{Z};\boldsymbol{\theta}\right)\boldsymbol{\Psi}'\left(\mathbf{Z};\boldsymbol{\theta}\right)\right]. \end{aligned}$$

For an ordinary M-estimate,

$$\boldsymbol{\Psi}\left(\mathbf{z};\boldsymbol{\theta}\right) = \psi\left(\frac{y - \mathbf{x}'\boldsymbol{\theta}}{\sigma}\right)\mathbf{x}_i,$$

and with $E_{F_0}\left[\mathbf{x}\mathbf{x}'\right]$ estimated by

$$E_{\hat{F}_n}\left[\mathbf{x}\mathbf{x}'\right] = \frac{1}{n}\mathbf{X}'\mathbf{X},$$

this agrees with the result obtained earlier:

$$\mathbf{M} = \frac{1}{\sigma} E\left[\psi'\left(\frac{\varepsilon}{\sigma}\right)\right] \cdot \frac{1}{n}\mathbf{X}'\mathbf{X},$$
$$\mathbf{Q} = E\left[\psi^2\left(\frac{\varepsilon}{\sigma}\right)\right] \cdot \frac{1}{n}\mathbf{X}'\mathbf{X},$$
$$\hat{\boldsymbol{\theta}} \overset{d}{\approx} N\left(\boldsymbol{\theta}, V(\psi, F_0)\left(\mathbf{X}'\mathbf{X}\right)^{-1}\right).$$

- A proposal to modify the definition of an M-estimate, so as to bound the influence of outlying $\mathbf{x}$-values, resulted in 'Generalized M-estimation'. A GM-estimate is a solution to

  $$\frac{1}{n}\sum \eta\left(\mathbf{x}_i, \frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{\sigma}}\right)\mathbf{x}_i = \mathbf{0},$$

  where

  $$\eta\left(\mathbf{x}_i, \frac{r_i(\boldsymbol{\theta})}{\sigma}\right) = w(\mathbf{x}_i)\,\psi\left(\frac{r_i(\boldsymbol{\theta})}{\sigma}\right).$$

  (There are other variations of this in the literature). The weights $w(\mathbf{x}_i)$ are to be chosen for robustness against outlying $\mathbf{x}$-values. As with (ordinary) M-estimates, scale is estimated by solving an auxiliary equation. A GM-estimate can be computed just as an M-estimate was, by IRLS. Alternatively, use Newton-Raphson: define

$$\mathbf{G}\left(\hat{\boldsymbol{\theta}}\right) = \frac{1}{n}\sum \eta\left(\mathbf{x}_i, \frac{y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}}{\hat{\sigma}}\right)\mathbf{x}_i$$

and solve $\mathbf{G}\left(\hat{\boldsymbol{\theta}}\right) = \mathbf{0}$ through the iteration scheme

$$\boldsymbol{\theta}_{(k+1)} = \boldsymbol{\theta}_{(k)} - \left[\dot{\mathbf{G}}\left(\boldsymbol{\theta}_{(k)}\right)\right]^{-1}\mathbf{G}\left(\boldsymbol{\theta}_{(k)}\right)$$

with

$$\dot{\mathbf{G}}\left(\boldsymbol{\theta}_{(k)}\right) = \frac{-1}{n\hat{\sigma}}\sum \eta'\left(\mathbf{x}_i, \frac{y_i - \mathbf{x}_i'\boldsymbol{\theta}_{(k)}}{\hat{\sigma}}\right)\mathbf{x}_i\mathbf{x}_i'$$

(where $\eta'(\mathbf{x}, r) = (d/dr)\,\eta(\mathbf{x}, r) = w(\mathbf{x})\,\psi'(r)$).

- As before the estimate is asymptotically normal:

$$\sqrt{n}\left(\hat{\boldsymbol{\theta}}_{GM} - \boldsymbol{\theta}\right) \xrightarrow{L} N\left(\mathbf{0}, \mathbf{M}^{-1}\mathbf{Q}\mathbf{M}^{-1}\right); \tag{20.2}$$

the calculations now use

$$\boldsymbol{\Psi}(\mathbf{z}; \boldsymbol{\theta}) = w(\mathbf{x})\,\psi\left(\frac{y - \mathbf{x}'\boldsymbol{\theta}}{\sigma}\right)\mathbf{x}.$$

Thus

$$\begin{aligned}\mathbf{M} &= E\left[-\frac{\partial}{\partial\boldsymbol{\theta}}\boldsymbol{\Psi}\left(\mathbf{Z};\boldsymbol{\theta}\right)\right]\\ &= \frac{1}{\sigma}E\left[\psi'\left(\frac{\varepsilon}{\sigma}\right)\right]E\left[w\left(\mathbf{x}\right)\mathbf{x}\mathbf{x}'\right],\\ \mathbf{Q} &= E\left[\boldsymbol{\Psi}\left(\mathbf{Z};\boldsymbol{\theta}\right)\boldsymbol{\Psi}'\left(\mathbf{Z};\boldsymbol{\theta}\right)\right]\\ &= E\left[\psi^2\left(\frac{\varepsilon}{\sigma}\right)\right]E\left[w^2\left(\mathbf{x}\right)\mathbf{x}\mathbf{x}'\right].\end{aligned}$$

These are estimated by replacing the expectations by averages over the sample. With $V\left(\psi,F\right)$ as at (18.1) and $\mathbf{W}$ the diagonal matrix of weights, the result is that $\hat{\boldsymbol{\theta}}_{GM} \overset{d}{\approx} N\left(\boldsymbol{\theta},\boldsymbol{\Sigma}\left(F\right)\right)$, with

$$\boldsymbol{\Sigma}\left(F\right) = V\left(\psi,F\right)\cdot\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\left(\mathbf{X}'\mathbf{W}^2\mathbf{X}\right)\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}.$$

- **High breakdown estimators**. In can be shown that the BP of a GM-estimate is only about $1/p$. An early attempt at finding a regression estimate with very high BP led to the 'Least Median of Squares' estimate. This is defined by

$$med\left\{\left(y_i - \mathbf{x}_i'\hat{\boldsymbol{\theta}}\right)^2\right\} = \min.$$

Formally, if the absolute values of the residuals are ordered: $|r|_{(1)} \leq \cdots \leq |r|_{(n)}$, then

$$med\left\{|r|^2_{(i)}\right\} = \text{min}\,.$$

The LMS estimate is in general very difficult to compute (more on this later), does not have a limiting Normal distribution, and in fact converges to a non-Normal distribution at the rate $n^{-1/3}$, i.e. more slowly than the usual $n^{-1/2}$. But $BP = 1/2$.

- A more recent proposal is 'Least Trimmed Squares'. The LTS regression method minimizes the sum of the h smallest squared residuals, where h must be at least half the number of observations and is typically taken to be slightly greater than $n/2$. Formally,

$$\sum_{i=1}^{h} |r|^2_{(i)} = \text{min}\,.$$

Again difficult to compute, but it converges at the standard rate of $n^{-1/2}$ and has a BP of .5. A drawback is that it is very inefficient if the errors are in fact Normal.

21. One-step GM-estimation

- A drawback of GM-estimation is that the BP, while positive, is only about $1/p$. A way out of the problem is to compute a 'one-step' GM-estimate:

  1. Take a high breakdown initial estimate of $\boldsymbol{\theta}$, such as the LTS estimate, and a corresponding scale estimate $\hat{\sigma} = \sqrt{\frac{1}{h}\sum_{i=1}^{h} |r|_{(i)}^2}$. (This is multiplied by a correction factor – see ltsReg(robustbase) or the Pison, Van Aelst & Willems paper on the course website for details.)

  2. Compute as well highly robust weights $w(\mathbf{x}_i)$ (discussed later).

  3. Perform just one iteration of Newton-Raphson (not IRLS – this results in the wrong asymptotic properties when only one iteration is performed). Update $\hat{\sigma}$ to $\hat{\sigma} = MAD$.

- It can be shown – see the Simpson, Ruppert & Carroll paper on the course website for details – that $\hat{\boldsymbol{\theta}}$ computed in this way inherits the high BP of the initial estimate, while gaining the high efficiency of the M-estimate. In particular (20.2) continues to hold.

- One need not stop at one step; one can use $\boldsymbol{\theta}_{(1)}$ in place of $\hat{\boldsymbol{\theta}}_{LTS}$ and do one more iteration, obtaining a two-step GM estimate $\boldsymbol{\theta}_{(2)}$, etc. In fact SR&C recommend a three-step. BUT the number of iterations k must be decided on in advance, otherwise k becomes the value of a r.v. K and the asymptotic properties change.

- How can the robust weights $w(\mathbf{x})$ be computed? These should decrease as $\mathbf{x}$ moves away from the rest of the sample. An obvious possibility is $w(\mathbf{x}_i) = 1 - h_{ii}$, but these are very non-robust – outlying $\mathbf{x}_i$ can determine the measure (the 'masking' effect). This is most clear in straight line regression, where

$$h_{ii} = \frac{1}{n} + \frac{(x_i - \bar{x})^2}{\sum \left(x_j - \bar{x}\right)^2}.$$

- To get more robust weights, we first look for robust estimates $\mathbf{t}$ and $\mathbf{V}$ of the location and scatter of the $\mathbf{x}_i$. The Minimum Covariance Determinant (MCD) method finds the h ($> n/2$) observations $\mathbf{x}_{(i)}$ whose classical covariance matrix

  $$\mathbf{V} = \frac{1}{h} \sum_i \left(\mathbf{x}_{(i)} - \mathbf{t}\right) \left(\mathbf{x}_{(i)} - \mathbf{t}\right)'$$

  (here $\mathbf{t} = \bar{\mathbf{x}}$, the average of these h points) has the lowest possible determinant. Then a reweighting step is carried out to improve the efficiency – see covMcd(robustbase) or the Pison, Van Aelst & Willems paper for details. This results in robust estimates $\hat{\boldsymbol{\mu}}$ (a re-weighted average of these h points) and $\hat{\boldsymbol{\Sigma}}$ (a reweighted covariance matrix).

Finally, weights are computed:

$$w(\mathbf{x}_i) = \text{min}\left(1, \frac{\chi^2_{p-1}(.95)}{(\mathbf{x}_i - \hat{\boldsymbol{\mu}})' \hat{\boldsymbol{\Sigma}}^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})}\right)^{1/2}.$$

- The original proposal of SR&C was to use Minimum Volume Ellipsoid (MVE) weights. This method looks for the ellipsoid

$$\left\{\mathbf{x} \,|\, (\mathbf{x} - \mathbf{t})' \mathbf{V}^{-1} (\mathbf{x} - \mathbf{t}) \leq 1\right\}$$

  of smallest volume, subject to the requirement that it contain at least half of the data points. This suffers from the same problems as the LMS estimate, however.

- An R function to compute K-step GM-estimates is on the course website. It is applied here to the ‘mineral’ data set. See plots below. The weights $w(\mathbf{x}_i)$ are:

```
> print(round(GMfit$wx,2))
 [1] 1.00 1.00 0.43 0.68 0.51 1.00 1.00 1.00 1.00
[10] 1.00 1.00 1.00 1.00 1.00 0.16 0.44 0.43 0.66
[19] 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00
[28] 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00
[37] 1.00 1.00 0.61 1.00 1.00 1.00 1.00 1.00 1.00
[46] 1.00 1.00 1.00 1.00 0.71 1.00 1.00 1.00
```

[natheight=18.2693cm, natwidth=18.2693cm, height=11.332
width=12.3362cm] C:/sw50/temp/graphics/mineral2$_{32.pdf}$

- The remaining lines, very close to the LS line (after removing point 15) are the GM line using a (redescending) 'bisquare' ψ-function

  $$\psi_{bi}\left(r;c\right) = r\left(1-\left(\frac{r}{c}\right)^2\right)^2 I\left(|r|\leq c\right)$$

  with $c = 4.5$, and the MM line, to be considered next.

```
# Load the robustbase package first:
library(robustbase)

myGM = function(x, y, c, intercept, wts, K) {
... housekeeping stuff ...

# Define the robust weights
w = function(x) {
qwe = covMcd(x)
if(ncol(x)>1) mah = qwe$mah else
mah = (x-rep(qwe$center,length(x)))^2
        /as.numeric(qwe$cov)
if(wts=="w1") weights = sqrt(pmin(1,
  qchisq(.95, ncol(x))/mah)) else
if(wts=="w2") weights = pmax(0, (1-(mah/
  qchisq(.95, ncol(x)))^3)^3)
 #Alternate weights, cutting influence
  of outlying xs to zero
weights}

# eta and etaprime
eta = function(wx,r) wx*psi(r)
etaprime = function(wx,r) wx*psiprime(r)

# Start with the LTS estimate
init.fit = ltsReg(x,y, int = intercept)
theta = init.fit$coef
r = init.fit$resid
sigma = init.fit$scale #Initial scale estimate
std.res = r/sigma
wx = w(x)
out[1,] = round(c(theta, sigma),5)

for  (k in 1:K) {
G = t(X)%*%eta(wx,std.res)/n
Gdot = (-1/(n*sigma))*t(X)%*%
 (as.vector(etaprime(wx,std.res))*X)
Gdot.qr = qr(Gdot)
theta = theta-qr.solve(Gdot.qr,G)
r = y-X%*%theta
sigma = mad(r, center=0)
std.res = r/sigma
out = rbind(out, round(c(theta, sigma),5))
  }
list(out=out, theta=theta, wx=wx,
   std.res = std.res, coef = theta, sigma = sigma)}
```

22. MM-estimation

- MM estimation (so named because it uses two M-estimates) is the high breakdown regression method currently in vogue. Like GM-estimation it starts with a high breakdown initial estimate. But rather than LTS or LMS another method – 'S estimation' – is used and is discussed below.

- The method depends on two bounded 'ρ-functions' ρ_0 and ρ_1. Such functions must be nondecreasing in $|r|$, with $\rho(0) = 0$, $\rho(\infty) = 1$ and ρ strictly increasing in $|r|$ where $\rho(r) < 1$. (So how must ψ look?). Recommended is the bisquare ρ-function

$$\rho_{bi}(r;c) = \min\left\{1, 1 - \left(1 - \left(\frac{r}{c}\right)^2\right)^3\right\},$$

with derivative

$$\rho'_{bi}(r;c) = \frac{6}{c^2}\psi_{bi}(r;c).$$

- The S-estimate $\hat{\boldsymbol{\theta}}_S$, and scale estimate $\hat{\sigma}_S$, are defined as follows. For any $\hat{\boldsymbol{\theta}}$, with residuals $r_i\left(\hat{\boldsymbol{\theta}}\right)$, define a scale estimate $\hat{\sigma} = \hat{\sigma}\left(\hat{\boldsymbol{\theta}}\right)$ by

$$\frac{1}{n}\sum \rho_0\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}}\right) = .5. \tag{22.1}$$

  (Non-robust example: $\rho_0(r) = r^2/2$ gives $\hat{\sigma}^2 = \sum r_i^2/n$.) The S-estimate of regression is the solution to

$$\hat{\sigma}\left(\hat{\boldsymbol{\theta}}_S\right) = \min \hat{\sigma}\left(\hat{\boldsymbol{\theta}}\right) \tag{22.2}$$

  and then

$$\hat{\sigma}_S = \hat{\sigma}\left(\hat{\boldsymbol{\theta}}_S\right).$$

  This is the major computational challenge and is discussed below.

- Theoretical details are in the paper by Victor Yohai, on the course website.

- Regression is then estimated by solving

$$L\left(\hat{\boldsymbol{\theta}}\right) \overset{def}{=} \frac{1}{n}\sum \rho_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}_S}\right) = \text{min}, \tag{22.3}$$

starting with $\hat{\boldsymbol{\theta}}_S$. It is required that

$$\text{(a) } \rho_1 \leq \rho_0 \text{ and (b) } L\left(\hat{\boldsymbol{\theta}}\right) \leq L\left(\hat{\boldsymbol{\theta}}_S\right); \tag{22.4}$$

these ensure the high BP (→ .5). The rough idea is that (22.4) implies

$$\frac{1}{n}\sum \rho_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}_S}\right) \overset{(22.4b)}{\leq} \frac{1}{n}\sum \rho_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}_S\right)}{\hat{\sigma}_S}\right)$$
$$\overset{(22.4a)}{\leq} \frac{1}{n}\sum \rho_0\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}_S\right)}{\hat{\sigma}_S}\right) \overset{(22.1)}{=} .5;$$

thus not too many of the terms $r_i\left(\hat{\boldsymbol{\theta}}\right)/\hat{\sigma}_S$ can get large and the estimates must remain bounded.

- Asymptotically, any critical point arising from (22.3) will work. One proceeds, as before, by IRLS: write (22.3) as

$$\begin{aligned}\mathbf{0} &= \hat{\sigma}_S \cdot \frac{1}{n}\sum \psi_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}_S}\right)\mathbf{x}_i \\ &= \frac{1}{n}\sum w_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}\right)}{\hat{\sigma}_S}\right) r_i\left(\hat{\boldsymbol{\theta}}\right)\mathbf{x}_i\end{aligned}$$

with weights

$$w_1(r) = \frac{\psi_1(r)}{r};$$

repeatedly update $\boldsymbol{\theta}_{(j)}$ to

$$\boldsymbol{\theta}_{(j+1)} = \left(\mathbf{X}'\mathbf{W}_{(j)}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}_{(j)}\mathbf{y}.$$

The limit of this process is $\hat{\boldsymbol{\theta}}_{MM}$. It can be shown that $L\left(\hat{\boldsymbol{\theta}}\right)$ decreases at each step, so that 22.4b is guaranteed.

- The recommended ρ-functions are

$$\rho_0(r) = \rho_{bi}(r; c_0) \text{ and } \rho_1(r) = \rho_{bi}(r; c_1),$$

where:

1. $c_0 = 1.56$ so that, asymptotically for Normal errors, $\hat{\sigma}$ will correspond to the standard deviation: if $\varepsilon/\sigma_\varepsilon \sim N(0,1)$ then
$$E\left[\rho_{bi}\left(\frac{\varepsilon}{\sigma_\varepsilon}; c_0 = 1.56\right)\right] = .5;$$

2. c_1 must be $\geq c_0$ to satisfy 22.4(a), and is chosen for a prescribed efficiency at the Normal, e.g. for 95% efficiency relative to the LSE (with variance $\sigma_\varepsilon^2 \left(\mathbf{X}'\mathbf{X}\right)^{-1}$) the MM estimate (with variance $V\left(\psi_{bi}(\cdot; c_1), \Phi\right)\left(\mathbf{X}'\mathbf{X}\right)^{-1}$) should satisfy
$$.95 = \frac{\sigma_\varepsilon^2}{V\left(\psi_{bi}(\cdot; c_1), \Phi\right)} = \frac{\left\{E\left[\psi_{bi}'\left(\frac{\varepsilon}{\sigma_\varepsilon}; c_1\right)\right]\right\}^2}{E\left[\psi_{bi}^2\left(\frac{\varepsilon}{\sigma_\varepsilon}; c_1\right)\right]}.$$
This gives $c_1 = 4.68$; larger values give greater efficiency but allow large residuals to have a greater influence on the fit.

- Computation of $\hat{\boldsymbol{\theta}}_S$ and $\hat{\sigma}_S$: all approaches rely on 'subsampling' schemes; these are also used in the computation of LMS, MCD, etc. Consider a subsample

  $$\{(\mathbf{x}_i, y_i) \,|\, i \in J\},$$

  where J is any one of the $\binom{n}{p}$ sets of p indices chosen from $\{1, 2, \cdots, n\}$. Assume that the corresponding design matrix $\mathbf{X}_J$, with rows $\left\{\mathbf{x}_i' \,|\, i \in J\right\}$ has full rank (if not, drop this subsample and take another). Then $\mathbf{X}_J$, which is *square*, is invertible:

  $$\left(\mathbf{X}_J'\mathbf{X}_J\right)^{-1}\mathbf{X}_J' = \mathbf{X}_J^{-1}.$$

  (This might be the only time that you will see "$\mathbf{X}^{-1}$" used correctly in this course!) The corresponding regression coefficients are

  $$\hat{\boldsymbol{\theta}}_J = \mathbf{X}_J^{-1}\mathbf{y}_J$$

  and this estimated model fits these p datapoints *exactly*:

  $$\hat{\mathbf{y}}_J = \mathbf{X}_J\hat{\boldsymbol{\theta}}_J = \mathbf{y}_J.$$

- Starting with any $\hat{\boldsymbol{\theta}}_J$, we can find iterates $\left(\hat{\boldsymbol{\theta}}_J^{(k)}, \hat{\sigma}_J^{(k)}\right)$ for which:
  (i) (22.1) is satisfied, and
  (ii) $\hat{\sigma}_J^{(k)}$ decreases at each step.
  Thus $\lim_{k\to\infty}\left(\hat{\boldsymbol{\theta}}_J^{(k)}, \hat{\sigma}_J^{(k)}\right)$ – call it $\left(\hat{\boldsymbol{\theta}}_{J,C}, \hat{\sigma}_{J,C}\right)$ and note that $\hat{\sigma}_{J,C} = \hat{\sigma}\left(\hat{\boldsymbol{\theta}}_{J,C}\right)$ – is at least a *local* minimum of the function $\hat{\sigma}\left(\hat{\boldsymbol{\theta}}\right)$.

- The final 'solution' $\left(\hat{\boldsymbol{\theta}}_S, \hat{\sigma}_S\right)$ is (ideally) approximated by the best of the $\left(\hat{\boldsymbol{\theta}}_{J,C}, \hat{\sigma}_{J,C}\right)$. In practice we can't consider all $\binom{n}{p}$ subsamples. Instead, a large number N of them are randomly chosen, and the best of the $\left(\hat{\boldsymbol{\theta}}_{J,C}, \hat{\sigma}_{J,C}\right)$, arising with these, is taken as the solution. See the paper by Salibian-Barrera & Yohai for more details and computational improvements.

- The algorithm to determine $\hat{\sigma}_{J,C}$ is as follows. Put $\hat{\boldsymbol{\theta}}_J^{(0)} = \hat{\boldsymbol{\theta}}_J$. For $k = 0, 1, \cdots$ to convergence:

1. Solve (22.1): $\frac{1}{n}\sum_{i=1}^n \rho_0\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}_J^{(k)}\right)}{\hat{\sigma}}\right) = .5$, obtaining $\hat{\sigma}_J^{(k)}$(which will be $\leq \hat{\sigma}_J^{(k-1)}$). This can be done by introducing weights $w_0(r) = \rho_0\left(r\right)/r^2$ and iterating:
$$\hat{\sigma}^2 \leftarrow \frac{2}{n}\sum w_0\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}_J^{(k)}\right)}{\hat{\sigma}}\right) r_i\left(\hat{\boldsymbol{\theta}}_J^{(k)}\right)^2$$
to convergence, starting with $\hat{\sigma}_J^{(k-1)}$. (What is $\hat{\sigma}_J^{(0)}$? Does it matter?)

2. Do one step of WLS to get
$$\hat{\boldsymbol{\theta}}_J^{(k+1)} = \left(\mathbf{X}'\mathbf{W}_{(k)}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{W}_{(k)}\mathbf{y},$$
with weights $w_0\left(r_i\left(\hat{\boldsymbol{\theta}}_J^{(k)}\right)/\hat{\sigma}_J^{(k)}\right)$.

- Inferences (conditional on x) can be made in the same manner as described earlier; again a final least squares regression on pseudovalues gives an asymptotically correct printout. This however does not account for the possible lack of robustness in estimating $E\left[\mathbf{x}\mathbf{x}'\right]$ by $\mathbf{X}'\mathbf{X}/n$.

- On R: `mmfit = lmrob(y~x)` (after loading the robustbase library). The output will include a robust estimate (`mmfit$cov`) of
$$cov\left[\hat{\boldsymbol{\theta}}|\mathbf{X}\right] = V\left(\psi, F\right)\left(\mathbf{X}'\mathbf{X}\right)^{-1},$$
with $\mathbf{X}'\mathbf{X}/n$ replaced by the more robust
$$\frac{1}{\sum w_i}\sum w_i\mathbf{x}_i\mathbf{x}_i' \tag{22.5}$$
where $w_i = w_1\left(\frac{r_i\left(\hat{\boldsymbol{\theta}}_{MM}\right)}{\hat{\sigma}_S}\right)$. The idea here is that (22.5) converges to
$$\frac{1}{E\left[w_1\left(\varepsilon/\sigma_\varepsilon\right)\right]}E\left[w_1\left(\varepsilon/\sigma_\varepsilon\right)\mathbf{x}\mathbf{x}'\right] \overset{how?}{=} E\left[\mathbf{x}\mathbf{x}'\right].$$
Then the usual normal-theory inferences can be made, and are asymptotically correct.

- Example: An MM-fit to the stackloss data, followed by the t-test of the hypothesis that Acid.Conc. can be dropped, gives a point estimate of $\hat{\theta}_4 = -.113$ and a p-value of .125. Compare with the output of Lecture 18 - the ordinary M-estimate gave $\hat{\theta}_4 = -.12$ and a p-value of .325.

```
mmfit = lmrob(y~x)
theta = mmfit$coef
V = mmfit$cov
t.acid = theta[4]/sqrt(V[4,4])
p.acid = 2*(1-pt(abs(t.acid),n-p))
```

- Note: Despite its apparent complexity, an MM-estimate is just an ordinary M-estimate corresponding to a redescending ψ. The complexity comes in only through the initial estimates $\left(\hat{\sigma}_S, \hat{\boldsymbol{\theta}}_S\right)$; then their high BP is inherited by virtue of (22.4).

Part V

Design

23. Classical regression designs

- We suppose that the experimenter is able to choose the points $\mathbf{x}_i$ at which to observe Y. A change of notation is convenient – write the usual regression model now as
$$Y_i = \mathbf{z}'(\mathbf{x}_i)\boldsymbol{\theta} + \varepsilon_i,$$
where the $\mathbf{x}_i$ are the values of the independent *variables* (chosen by the experimenter) and the $\mathbf{z}(\mathbf{x}_i)$ are the *regressors*. For example in straight line regression: $Y_i = \theta_0 + \theta_0 x_i + \varepsilon_i$, the experimenter chooses the x_i; the regressors are $\mathbf{z}'(x_i) = (1, x_i)$.

- If the model can be trusted to be correct, then the LSEs are unbiased and variance minimization is the goal. If $\mathbf{Z}$ has rows $\{\mathbf{z}'(x_i)\}_{i=1}^n$ then
$$\text{cov}\left[\hat{\boldsymbol{\theta}}\right] = \sigma_\varepsilon^2\left(\mathbf{Z}'\mathbf{Z}\right)^{-1},$$
and are to choose $\{\mathbf{x}_i\}_{i=1}^n$ to minimize some scalar-valued function of $(\mathbf{Z}'\mathbf{Z})^{-1}$.

- Examples:

  – det $\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right]$. A confidence ellipsoid on $\boldsymbol{\theta}$ is

  $$\mathbb{E}\left(c^2\right) = \left\{\boldsymbol{\theta} \mid \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)' \mathbf{Z}'\mathbf{Z}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right) \le c^2\right\},$$

  with $c^2 = S^2 p F^p_{n-p}(1-\alpha)$. The volume is $\int_{\mathbb{E}(c^2)} d\boldsymbol{\theta}$; with the QR-decomposition $\mathbf{Z} = \mathbf{QR}$ and $\mathbf{t} = \mathbf{R}\left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}\right)/c$ this becomes

  $$\begin{aligned} vol &= \int_{\|\mathbf{t}\|\le 1} \left|\left(\frac{\partial \boldsymbol{\theta}}{\partial \mathbf{t}}\right)\right|_+ d\mathbf{t} \\ &= \int_{\|\mathbf{t}\|\le 1} \left|c\mathbf{R}^{-1}\right| d\mathbf{t} \\ &= c^p \left|\mathbf{R}^{-1}\right| \int_{\|\mathbf{t}\|\le 1} d\mathbf{t} \\ &= c^p \left|\mathbf{Z}'\mathbf{Z}\right|^{-1/2} \cdot vol\,(\text{unit sphere in } \mathbb{R}^p), \end{aligned}$$

  so minimizing det $\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right]$ – equivalently, maximizing det $[\mathbf{Z}'\mathbf{Z}]$ – results in a confidence ellipsoid of minimum volume. This is called the ‘D-optimality’ design problem.

- $tr\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right]$ – minimizing this is the 'A-optimality' design problem. The variances of the $\hat{\theta}_j$ are proportional to the diagonal elements of $(\mathbf{Z}'\mathbf{Z})^{-1}$, so an A-optimal design results in the smallest value of the average of these variances.

- $ch_{\max}\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right]$ – minimizing this maximum eigenvalue is the 'E-optimality' problem. The motivation is that

$$\begin{aligned}\max_{\|\mathbf{c}\|\leq k} \text{var}\left[\mathbf{c}'\hat{\boldsymbol{\theta}}\right] &= \max_{\|\mathbf{c}\|\leq k}\left[\sigma_\varepsilon^2\cdot\mathbf{c}'\left(\mathbf{Z}'\mathbf{Z}\right)^{-1}\mathbf{c}\right]\\ &= \sigma_\varepsilon^2 k^2 \max_{\|\mathbf{c}\|\leq 1}\mathbf{c}'\left(\mathbf{Z}'\mathbf{Z}\right)^{-1}\mathbf{c}\\ &= \sigma_\varepsilon^2 k^2 ch_{\max}\left[\left(\mathbf{Z}'\mathbf{Z}\right)^{-1}\right],\end{aligned}$$

so that this maximum variance is minimized by the E-optimal design.

- Straight line regression, $-1 \leq x \leq 1$, n even.

$$\left(\mathbf{Z}'\mathbf{Z}\right)^{-1} = \frac{1}{S_{xx}}\begin{pmatrix} m_2 & -\bar{x}\\ -\bar{x} & 1\end{pmatrix}, \text{ with}$$

$$S_{xx} = \sum\left(x_i - \bar{x}\right)^2 \text{ and } m_2 = \frac{\sum x_i^2}{n} = \frac{S_{xx}}{n} + \bar{x}^2.$$

Then:

1. det $\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right] = (nS_{xx})^{-1}$ is minimized by maximizing S_{xx}, leading to ... with $S_{xx} = n$.

2. $tr\left[(\mathbf{Z}'\mathbf{Z})^{-1}\right] = \frac{1+m_2}{S_{xx}} = \frac{1}{n} + \frac{1+\bar{x}^2}{S_{xx}} \geq \frac{1}{n} + \frac{1}{S_{xx}}$ is minimized by maximizing S_{xx} ... .

3. E-optimality leads to the same design.

- Robustness issues? Testing Lack of Fit?

- Deriving optimal designs for other models can be much more difficult; this is an active and exciting field of research.

- **Response surface methodology**. Here a common goal is to determine where, on a surface of interest, the response is a maximum. For instance suppose $Y = f(x_1, x_2) + \varepsilon$ and we want to determine where $f(\cdot, \cdot)$ (whose precise form is unknown) is a maximum. First assume the variables have been coded, so that $-1 \leq x_1, x_2 \leq 1$. Fit a linear model

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_1 + \hat{\theta}_2 x_2$$

and plot the lines $\hat{y} = k$ for increasing k; this gives the *path of steepest ascent*. (Here one might use a design with points at the corners of the square – this is D-optimal – and a few scattered throughout the square for robustness.) Next move along the path of steepest ascent, starting at $(0, 0)$, taking observations along the way until y starts to decrease; this gives the approximate location, say $\left(x_1^*, x_2^*\right)$, of the maximum (although more exploration might be needed; this should be indicated by the linear models being fitted along the way).

Finally, take observations in a neighbourhood of $\left(x_1^*, x_2^*\right)$ and fit a quadratic model

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_1 + \hat{\theta}_2 x_2 + \hat{\theta}_{11} x_1^2 + \hat{\theta}_{22} x_2^2 + \hat{\theta}_{12} x_1 x_2;$$

do the calculus to obtain the stationary point.

- How should this final design, to estimate the quadratic model, be constructed? First recode the variables so that, again, $-1 \leq x_1, x_2 \leq 1$. We might first require that the predictions from the fit all have the same variance, for points $\mathbf{x} = (x_1, x_2)'$ equidistant from the centre of the design region. In other words, with $\mathbf{z}(\mathbf{x}) = \left(1, x_1, x_2, x_1^2, x_2^2, x_1 x_2\right)'$, we have that

$$\mathbf{z}'(\mathbf{x}) \left(\mathbf{Z}'\mathbf{Z}\right)^{-1} \mathbf{z}(\mathbf{x})$$

depends on $\mathbf{x}$ only through $\|\mathbf{x}\|$. Such a design is called *rotatable*. One way to construct a rotatable design for this problem is to start with a *central composite design*, with one point at each of the corners, one or more at $(0, 0)$, and one at each of $(\pm\alpha, 0)$ and $(0, \pm\alpha)$. Then determine α $(= \sqrt{2})$ for rotatability.

- **Designs for nonlinear regression**. The model is $\mathbf{Y} = \boldsymbol{\eta}(\boldsymbol{\theta}) + \boldsymbol{\varepsilon}$, in which $\boldsymbol{\eta}(\boldsymbol{\theta})$ has elements $f(\boldsymbol{\theta}; \mathrm{x}_i)$ and derivative $\mathbf{V}(\boldsymbol{\theta}) = \partial\boldsymbol{\eta}/\partial\boldsymbol{\theta}$. The estimate is approximately normal:

$$\hat{\boldsymbol{\theta}} \overset{d}{\approx} N\left(\boldsymbol{\theta}, \sigma_\varepsilon^2 \left(\mathbf{V}'\mathbf{V}\right)^{-1}\right),$$

and so we might aim to minimize det $\left[\left(\mathbf{V}'\mathbf{V}\right)^{-1}\right]$ (which depends on the unknown $\boldsymbol{\theta}$). We do so by *maximizing* det $\left[\mathbf{V}'\mathbf{V}\right]$. There are several possibilities:

1. Maximin approach: choose the design points $\{\mathrm{x}_i\}$ so as to maximize $\min_{\boldsymbol{\theta}} \left|\mathbf{V}'\mathbf{V}\right|$. This might be overly pessimistic.

2. Bayesian approach: maximize

$$\int \left|\mathbf{V}'(\boldsymbol{\theta})\,\mathbf{V}(\boldsymbol{\theta})\right| p(\boldsymbol{\theta})\, d\boldsymbol{\theta}$$

for some *prior density* $p(\boldsymbol{\theta})$.

3. Another suggestion (Box and Lucas) is to first take $n = p$, so that $\mathbf{V}(\boldsymbol{\theta})$ is square, and choose p points $\{\mathbf{x}_i\}$ so as to maximize $|\mathbf{V}'\mathbf{V}| = |\mathbf{V}|^2$. Typically, the D-optimal design consists of replicating these p points. But this method relies on the accuracy of our initial guess for $\boldsymbol{\theta}$.

4. A sequential approach, if possible, is preferable. Starting with, say, the Box-Lucas design with $n = p$, one adds points $\mathbf{x}_i$ sequentially so as to increase $|\mathbf{V}'\mathbf{V}|$ at each step.
   Suppose we already have an n-point design, resulting in $\mathbf{V}_n(\boldsymbol{\theta})$, and wish to add one more point $\mathbf{x}_{n+1}$, resulting in

$$\mathbf{V}_{n+1}(\boldsymbol{\theta}) = \begin{pmatrix} \mathbf{V}_n(\boldsymbol{\theta}) \\ \dot{\mathbf{f}}'_{n+1} \end{pmatrix},$$

where $\dot{\mathbf{f}}_{n+1}^{\prime} = \partial f\left(\boldsymbol{\theta}; \mathbf{x}_{n+1}\right)/\partial\boldsymbol{\theta}$. Then

$$\begin{aligned} & \left|\mathbf{V}_{n+1}^{\prime}\mathbf{V}_{n+1}\right| \\ = & \left|\mathbf{V}_{n}^{\prime}\mathbf{V}_{n} + \dot{\mathbf{f}}_{n+1}\dot{\mathbf{f}}_{n+1}^{\prime}\right| \\ = & \left|\mathbf{V}_{n}^{\prime}\mathbf{V}_{n}\right| \left|\mathbf{I} + \left(\mathbf{V}_{n}^{\prime}\mathbf{V}_{n}\right)^{-1}\dot{\mathbf{f}}_{n+1}\dot{\mathbf{f}}_{n+1}^{\prime}\right| \\ = & \left|\mathbf{V}_{n}^{\prime}\mathbf{V}_{n}\right| \left(1 + \dot{\mathbf{f}}_{n+1}^{\prime}\left(\mathbf{V}_{n}^{\prime}\mathbf{V}_{n}\right)^{-1}\dot{\mathbf{f}}_{n+1}\right); \end{aligned}$$

thus

$$\begin{aligned} \mathbf{x}_{n+1} &= \operatorname{arg\,max} \dot{\mathbf{f}}_{n+1}^{\prime}\left(\mathbf{V}_{n}^{\prime}\mathbf{V}_{n}\right)^{-1}\dot{\mathbf{f}}_{n+1} \\ &= \operatorname{arg\,max} \left\|\mathbf{R}^{-1\prime}\dot{\mathbf{f}}_{n+1}\right\|^{2}, \end{aligned}$$

in terms of the QR-decomposition of $\mathbf{V}_n$. The objective is evaluated at the current estimate $\hat{\boldsymbol{\theta}}$.

This is all quite straightforward numerically, and sometimes analytically as well.

- Example: The Michaelis-Menten model has $f(\alpha, \beta; x) = \alpha x / (\beta + x)$ and
$$\dot{\mathbf{f}}' = \left(\frac{x}{\beta + x}, -\frac{\alpha x}{(\beta + x)^2}\right).$$
A starting design with $n = 2$ and $0 \leq x_1 < x_2 \leq x_{\text{max}}$ is obtained by maximizing
$$|\mathbf{V}| = \frac{\alpha x_1 x_2 (x_2 - x_1)}{(\beta + x_1)^2 (\beta + x_2)^2} > 0.$$
It is easier to write $z_i = 1/x_i$ and maximize
$$\begin{aligned} v(z_1, z_2) &= \log(|\mathbf{V}| / \alpha) \\ = & \log(z_1 - z_2) - 2\log(1 + \beta z_1) - 2\log(1 + \beta z_2) \end{aligned}$$
over $z_{\text{min}} \leq z_2 < z_1 \leq \infty$. Since
$$\frac{\partial v}{\partial z_2} = \frac{-1 + \beta z_2 - 2\beta z_1}{(z_1 - z_2)(1 + \beta z_2)} < 0$$
for $z_2 < z_1$ we should choose z_2 as small as possible: $z_2 = z_{\text{min}}$. Then $v(z_1, z_{\text{min}})$ is found to be maximized by $z_1 = 2z_{\text{min}} + \beta^{-1}$. Thus the starting design is
$$x_1 = \frac{\beta}{1 + \frac{2\beta}{x_{\text{max}}}} \approx \beta, \; x_2 = x_{\text{max}},$$

evaluated at an initial guess β (the 'halfway' point). The design should not depend on the conditionally linear parameter α.

- After n observations have been made, and $\hat{\alpha}$, $\hat{\beta}$ and

  $$\mathbf{R}^{-1'} = \begin{pmatrix} r_1 & 0 \\ r_2 & r_3 \end{pmatrix}$$

  computed, the $(n+1)^{th}$ observation is obtained by maximizing

  $$\begin{aligned} & \left\| \mathbf{R}^{-1'} \dot{\mathbf{f}}_{n+1} \right\|^2 \\ = & \left(r_1 \frac{x}{\hat{\beta}+x} \right)^2 + \left(r_2 \frac{x}{\hat{\beta}+x} - r_3 \frac{\hat{\alpha} x}{\left(\hat{\beta}+x\right)^2} \right)^2 \\ = & z^2 \left[r_1^2 + \left\{ r_2 - r_3 \gamma \left(1-z\right) \right\}^2 \right] \Big|_{\substack{z = x/\left(\hat{\beta}+x\right) \\ \gamma = \hat{\alpha}/\hat{\beta}}} \end{aligned}$$

  for $0 \leq z \leq x_{\text{max}} / \left(\hat{\beta} + x_{\text{max}} \right) \leq 1$. Then $x_{n+1} = \hat{\beta} z / \left(1-z\right)$.

24. Robust regression designs I

- As with LSEs, designs which are optimal for a particular model tend to be good only when that model is exactly correct. Box and Draper (1959; on course website) study designs for polynomial fits, when the true response is a polynomial of higher degree than the one fitted. They compare designs ranging from the classically optimal (minimizing the variance; they have only as many support points as parameters being estimated), to the uniform (i.e., equally spaced design points; to minimize the bias). They conclude "... the optimal design in typical situations in which both variance and bias occur is very nearly the same as would be obtained if *variance were ignored completely* and the experiment designed so as to *minimize bias alone*."

- **Example**: Suppose that one estimates a straight line for $x \in [-1, 1]$, obtaining the LS estimate $\hat{\theta}_0 + \hat{\theta}_1 x$. Now suppose that the true response is quadratic: $E[Y|x] = \theta_0 + \theta_1 x + \theta_2 x^2$. Define the prediction bias at x by

$$bias\,(x) = E\left[\hat{\theta}_0 + \hat{\theta}_1 x\right] - \left\{\theta_0 + \theta_1 x + \theta_2 x^2\right\}.$$

Then (assigned) for a symmetric design, and with $m_2 = n^{-1}\sum x_i^2$, the *integrated squared bias* is

$$B = \int_{-1}^{1} bias^2\,(x)\,dx = 2\theta_2^2 \left\{\left(m_2 - \frac{1}{3}\right)^2 + \frac{4}{45}\right\}.$$

This is *maximized* by the D-optimal design ($m_2 = 1$) and minimized if $m_2 = 1/3$ ($=$ the second moment of the uniform distribution on $[-1, 1]$). One bias-minimizing design has equally spaced design points

$$x_i = -d + \frac{2d\,(i-1)}{n-1}, \text{ with } d = \sqrt{\frac{n-1}{n+1}}.$$

[natheight=17.7795cm, natwidth=17.7795cm, height=14.293
width=14.2934cm] C:/sw50/temp/graphics/slr$_{33.pdf}$

- Much has since been done on robustness of design; here is an outline of the development (see my Handbook of Design chapter *Robustness of Design* on the course web site).

- Suppose that one has a p-vector $\mathbf{z} = \mathbf{f}(\mathbf{x})$ of regressors, each element of which is a function of q variables $\mathbf{x} = (x_1, ..., x_q)'$, with $\mathbf{x}$ to be chosen, by the experimenter, from a finite *design space* $S = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$. Then the fitted model is $E[Y(\mathbf{x})] = \mathbf{f}'(\mathbf{x})\boldsymbol{\theta}$. The experimenter is concerned that, instead,
$$E[Y(\mathbf{x})] = \mathbf{f}'(\mathbf{x})\boldsymbol{\theta} + \psi(\mathbf{x}), \tag{24.1}$$
for some function ψ. There is an immediate problem concerning the interpretation of $\boldsymbol{\theta}$ (why?), this is avoided by first *defining* the target parameter by
$$\boldsymbol{\theta} = \arg\min_{\eta} \sum_{i=1}^{N} \left(E[Y(\mathbf{x}_i)] - \mathbf{f}'(\mathbf{x}_i)\eta\right)^2,$$

and then *defining*

$$\psi(\mathbf{x}) = E[Y(\mathbf{x})] - \mathbf{f}'(\mathbf{x})\boldsymbol{\theta};$$

this leads to the orthogonality requirement

$$\sum_{i=1}^{N} \mathbf{f}(\mathbf{x}_i)\,\psi(\mathbf{x}_i) = \mathbf{0}. \qquad (24.2)$$

- We measure the quality of a design through the

Average MSE of $\mathbf{f}'(\mathbf{x})\,\hat{\boldsymbol{\theta}}$ as an estimate of $E[Y(\mathbf{x})]$:

$$\begin{aligned}
\text{amse} &= \frac{1}{N}\sum_{i=1}^{N} E\left[\left(\mathbf{f}'(\mathbf{x}_i)\,\hat{\boldsymbol{\theta}} - E[Y(\mathbf{x}_i)]\right)^2\right] \\
&= \frac{1}{N}\sum_{i=1}^{N} E\left[\begin{pmatrix} \left\{\mathbf{f}'(\mathbf{x}_i)\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}\right)\right\} \\ -\left\{E[Y(\mathbf{x}_i)] - \mathbf{f}'(\mathbf{x}_i)\,\boldsymbol{\theta}\right\} \end{pmatrix}^2\right] \\
&= \frac{1}{N}\sum_{i=1}^{N} \mathbf{f}'(\mathbf{x}_i)\,E\left[\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}\right)\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}\right)'\right]\mathbf{f}(\mathbf{x}_i) \\
&\quad + \frac{1}{N}\sum_{i=1}^{N}\psi^2(\mathbf{x}_i) \\
&= tr\left\{\underbrace{\frac{1}{N}\sum_{i=1}^{N}\mathbf{f}(\mathbf{x}_i)\,\mathbf{f}'(\mathbf{x}_i)}_{=\ \mathbf{A}}\cdot\underbrace{E\left[\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}\right)\left(\hat{\boldsymbol{\theta}}-\boldsymbol{\theta}\right)'\right]}_{=\ \text{mse}\left[\hat{\boldsymbol{\theta}}\right]}\right\} + \\
&\quad \frac{1}{N}\sum_{i=1}^{N}\psi^2(\mathbf{x}_i) \\
&= tr\left\{\mathbf{A}\cdot\text{mse}\left[\hat{\boldsymbol{\theta}}\right]\right\} + \frac{1}{N}\sum_{i=1}^{N}\psi^2(\mathbf{x}_i)\,. \qquad (24.3)
\end{aligned}$$

Note that

$$\begin{aligned}\text{mse}\left[\hat{\boldsymbol{\theta}}\right] &= E\left[\left(\hat{\boldsymbol{\theta}} - E\left(\hat{\boldsymbol{\theta}}\right)\right)\left(\hat{\boldsymbol{\theta}} - E\left(\hat{\boldsymbol{\theta}}\right)\right)'\right] \\ &\quad + \left(E\left(\hat{\boldsymbol{\theta}}\right) - \boldsymbol{\theta}\right)\left(E\left(\hat{\boldsymbol{\theta}}\right) - \boldsymbol{\theta}\right)' \\ &= \text{cov}\left[\hat{\boldsymbol{\theta}}\right] + \left(\text{bias}\left[\hat{\boldsymbol{\theta}}\right]\right)\left(\text{bias}\left[\hat{\boldsymbol{\theta}}\right]\right)'.\end{aligned}$$

The covariance matrix of the lse depends only on the error variance and the regressors (not on the correctness of the model); if n_i of the n observations are to be made at $\mathbf{x}_i$ and $\xi_i = n_i/n$ it is

$$\text{cov}\left[\hat{\boldsymbol{\theta}}\right] = \sigma_\varepsilon^2 \left(\sum_{i=1}^{N} n_i \mathbf{f}'(\mathbf{x}_i)\,\mathbf{f}(\mathbf{x}_i)\right)^{-1} = \frac{\sigma_\varepsilon^2}{n}\mathbf{M}_\xi^{-1},$$

where

$$\mathbf{M}_\xi = \sum_{i=1}^{N} \xi_i \mathbf{f}(\mathbf{x}_i)\,\mathbf{f}'(\mathbf{x}_i).$$

If the n_i observations made at $\mathbf{x}_i$ are $\left\{Y_{ij}\right\}$ then

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \left(\frac{1}{n}\sum_{i=1}^{N} n_i \mathbf{f}\left(\mathbf{x}_i\right)\mathbf{f}'\left(\mathbf{x}_i\right)\right)^{-1}\frac{1}{n}\sum_{i=1}^{N}\sum_{j=1}^{n_i}\mathbf{f}\left(\mathbf{x}_i\right)Y_{ij};\\ E\left[\hat{\boldsymbol{\theta}}\right] &= \mathbf{M}_{\xi}^{-1}\cdot\frac{1}{n}\sum_{i,j}\mathbf{f}\left(\mathbf{x}_i\right)\left(\mathbf{f}'\left(\mathbf{x}_i\right)\boldsymbol{\theta}+\psi\left(\mathbf{x}_i\right)\right)\\ &= \mathbf{M}_{\xi}^{-1}\cdot\sum_{i=1}^{N}\xi_i\mathbf{f}\left(\mathbf{x}_i\right)\left(\mathbf{f}'\left(\mathbf{x}_i\right)\boldsymbol{\theta}+\psi\left(\mathbf{x}_i\right)\right)\\ &= \boldsymbol{\theta}+\mathbf{M}_{\xi}^{-1}\sum_{i=1}^{N}\xi_i\mathbf{f}\left(\mathbf{x}_i\right)\psi\left(\mathbf{x}_i\right).\end{aligned}$$

Thus, with $b_{\psi,\xi}=\sum_{i=1}^{N}\xi_i\mathbf{f}\left(\mathbf{x}_i\right)\psi\left(\mathbf{x}_i\right)$, we have

$$\begin{aligned}\text{bias}\left[\hat{\boldsymbol{\theta}}\right] &= \mathbf{M}_{\xi}^{-1}b_{\psi,\xi}\text{, hence}\\ \text{mse}\left[\hat{\boldsymbol{\theta}}\right] &= \frac{\sigma_{\varepsilon}^{2}}{n}\mathbf{M}_{\xi}^{-1}+\mathbf{M}_{\xi}^{-1}b_{\psi,\xi}b_{\psi,\xi}'\mathbf{M}_{\xi}^{-1}.\end{aligned}$$

Upon substituting into (24.3), we can write amse as

$$\frac{\sigma_{\varepsilon}^{2}}{n}tr\mathbf{A}\mathbf{M}_{\xi}^{-1}+b_{\psi,\xi}'\mathbf{M}_{\xi}^{-1}\mathbf{A}\mathbf{M}_{\xi}^{-1}b_{\psi,\xi}+\frac{1}{N}\sum_{i=1}^{N}\psi^{2}\left(\mathbf{x}_i\right);$$

we shall now call this $L\left(\psi,\xi\right)$.

- We seek a *minimax* design, which minimizes the maximum value of amse as ψ ranges over all functions satisfying (24.2) and (why?) $\sum_{i=1}^{N}\psi^{2}\left(\mathbf{x}_i\right) \leq \tau^2/n$ for a constant τ. Since amse increases if we can multiply ψ by a constant > 1, we assume that
$$\sum_{i=1}^{N}\psi^{2}\left(\mathbf{x}_i\right) = \tau^2/n. \qquad (24.4)$$
Then the problem is to find a vector $\xi = (\xi_1, \cdots, \xi_N)$, minimizing the maximum value of
$$L\left(\psi,\xi\right) = \frac{\sigma_\varepsilon^2}{n} tr\mathbf{A}\mathbf{M}_\xi^{-1} + b'_{\psi,\xi}\mathbf{M}_\xi^{-1}\mathbf{A}\mathbf{M}_\xi^{-1}b_{\psi,\xi} + \frac{\tau^2}{Nn},$$
subject to (24.2) and (24.4).

- Maximization over ψ. Define $\mathbf{F}$ to be the $N \times p$ matrix with rows $\mathbf{f}'\left(\mathbf{x}_i\right)$, $\boldsymbol{\psi}$ the vector with elements $\psi\left(\mathbf{x}_i\right)$, and $\mathbf{D}_\xi$ the diagonal matrix with diagonal elements ξ_i. Then
$$\mathbf{M}_\xi = \mathbf{F}'\mathbf{D}_{\boldsymbol{\xi}}\mathbf{F},\ b_{\psi,\xi} = \mathbf{F}'\mathbf{D}_{\boldsymbol{\xi}}\boldsymbol{\psi}, \mathbf{A} = N^{-1}\mathbf{F}'\mathbf{F}.$$
Apply the QR decomposition: $\mathbf{F} = (\mathbf{Q}_1 \vdots \mathbf{Q}_2)\begin{pmatrix}\mathbf{R}\\ \mathbf{0}\end{pmatrix} = \mathbf{Q}_1\mathbf{R}$, with $\mathbf{Q} = (\mathbf{Q}_1 \vdots \mathbf{Q}_2)$ orthogonal. Then (24.2)

becomes $\mathbf{Q}_1'\boldsymbol{\psi} = \mathbf{0}$, so that $\boldsymbol{\psi}$ is orthogonal to col$\left(\mathbf{Q}_1\right)$, hence is of the form $\boldsymbol{\psi} = \left(\tau/\sqrt{n}\right)\mathbf{Q}_2\mathbf{c}$. Then using (24.4), $1 = \|\mathbf{Q}_2\mathbf{c}\|^2 = \|\mathbf{c}\|^2$. After a calculation,

$$b_{\psi,\xi}'\mathbf{M}_\xi^{-1}\mathbf{A}\mathbf{M}_\xi^{-1}b_{\psi,\xi} + \frac{\tau^2}{Nn} = \frac{\tau^2}{n}\mathbf{c}'\mathbf{P}\mathbf{c}, \text{where}$$

$$\mathbf{P} = N^{-1}\left(\begin{array}{c} \mathbf{Q}_2'\mathbf{D}_\xi\mathbf{Q}_1\left(\mathbf{Q}_1'\mathbf{D}_\xi\mathbf{Q}_1\right)^{-2}\mathbf{Q}_1'\mathbf{D}_\xi\mathbf{Q}_2 \\ +\mathbf{I}_{N-p} \end{array}\right).$$

This is maximized subject to $\|\mathbf{c}\| = 1$ by choosing $\mathbf{c}$ to be the eigenvector of $\mathbf{P}$ corresponding to the maximum eigenvalue, and then $\mathbf{c}'\mathbf{P}\mathbf{c} = ch_{\max}\mathbf{P}$, so that

$$\max_\psi L\left(\psi,\xi\right) = \frac{\sigma_\varepsilon^2}{n}tr\mathbf{A}\mathbf{M}_\xi^{-1} + \frac{\tau^2}{n}ch_{\max}\mathbf{P}.$$

With $\nu = \tau^2/\left(\sigma_\varepsilon^2 + \tau^2\right)$, this is $\max_\psi L\left(\psi,\xi\right) = \frac{\sigma_\varepsilon^2+\tau^2}{n}$ times

$$L_\nu\left(\xi\right) = \left(1-\nu\right)tr\mathbf{A}\mathbf{M}_\xi^{-1} + \nu ch_{\max}\mathbf{P}.$$

The experimenter chooses $\nu \in [0,1]$ according to how much emphasis he/she wishes to place on bias reduction versus variance reduction, and need not know σ_ε^2 or τ^2.

- Returning to the original terms:

$$L_\nu\left(\boldsymbol{\xi}\right) = \left(1-\nu\right) tr\mathbf{A}\mathbf{M}_\xi^{-1} + \nu ch_{\mathsf{max}}\mathbf{K}_\xi\mathbf{H}_\xi^{-1},$$

where

$$\begin{aligned} \mathbf{A} &= N^{-1}\mathbf{F}'\mathbf{F}, \\ \mathbf{M}_\xi &= \mathbf{F}'\mathbf{D}_\xi\mathbf{F}, \\ \mathbf{K}_\xi &= \mathbf{F}'\mathbf{D}_\xi^2\mathbf{F}, \\ \mathbf{H}_\xi &= \mathbf{M}_\xi\mathbf{A}^{-1}\mathbf{M}_\xi. \end{aligned}$$

- The problem now is to find a vector $\boldsymbol{\xi} = \left(\xi_1 = \frac{n_1}{n}, \cdots, \xi_N = \frac{n_N}{n}\right)$ to minimize $L_\nu\left(\boldsymbol{\xi}\right)$. This problem is completely determined by the vectors $\mathbf{f}\left(\mathbf{x}\right)$, but depends very much on their structure. Each particular problem (SLR, quadratic regression, multiple linear regression with or without interactions, a linear approximation to a Michaelis-Menten response, etc.) has its own unique solution.

25. Robust regression designs II

- Minimization of $L_\nu(\boldsymbol{\xi})$ over N-vectors $\boldsymbol{\xi} = \left(\frac{n_1}{n}, \cdots, \frac{n_N}{n}\right)$. 'Genetic algorithms' are currently popular search methods through which one systematically improves the current designs so as to reduce the value of the loss. The idea is to mimic the evolution of biological populations.

- We start with a randomly chosen 'population' $\boldsymbol{\xi}_1, ..., \boldsymbol{\xi}_{40}$ of 40 designs. Compute the loss $L_{\nu,k} = L_\nu(\boldsymbol{\xi}_k)$ of each; this is the first 'generation'.

- Assign a 'fitness level' to each, with small loss corresponding to large fitness; normalize to get a probability distribution: first rank the $L_{\nu,k}$ from smallest to largest, then
$$\text{fitness}_k = \frac{1}{\sqrt{rank\left(L_{\nu,k}\right)}}; \ \psi_k = \frac{\text{fitness}_k}{\sum_k \text{fitness}_k}.$$
The designs with the smallest loss are assigned the highest probabilities ψ_k.

- Form the next generation:
  - The best (fittest) two in the current generation always survive to the next. A consequence is that the minimum loss in a generation can only decrease.
  - Otherwise, choose pairs of designs ('parents') from the current generation - $P\left(\text{choose } k^{th}\right) = \psi_k$ - and combine them to form a 'child'. Continue until a new generation of size 40 has been generated.

- Repeat, forming new generations and evaluating their fitnesses until the best design has not changed for 5000 consecutive generations.

- Parents combine by 'crossover' with probability $P_{crossover}(= 9)$; with probability $1 - P_{crossover}$ the child is identical to the fittest parent.

- Crossover: Represent a design $\boldsymbol{\xi}$ by the vector $n\boldsymbol{\xi}$; e.g. $n = 4, N = 5$; then parents combine as follows:

$$\max\left(\text{parents}\right) = \max\left\{\begin{pmatrix}0\\2\\1\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\0\\3\\1\end{pmatrix}\right\} = \begin{pmatrix}0\\2\\1\\3\\1\end{pmatrix}$$

$$\overset{\text{sum } = 7;}{\underset{\longrightarrow}{\text{randomly reduce}}} \begin{pmatrix}0\\2\\1\\2\\1\end{pmatrix} \rightarrow \begin{pmatrix}0\\2\\0\\2\\1\end{pmatrix} \rightarrow \begin{pmatrix}0\\1\\0\\2\\1\end{pmatrix} = \text{child}$$

- Mutation: In each child, randomly choose two elements, with probability $P_{mutation}$ ($=$ .05) swap them. Repeat N times with each child.

- The crossover and mutation mechanisms are generally quite arbitrary. The ‘tuning constants’ have here been chosen quite arbitrarily, and don’t seem to affect the performance much.

[natheight=7.7343cm, natwidth=16.5076cm, height=10.0715cm, width=14.1155cm]
C:/sw50/temp/graphics/cubic$_{34.pdf}$
Designs for cubic regression; design space is $N = 201$ equally spaced points spanning $[-1, 1]$. The D-optimal design places 1/4 of the observations at each of $\pm 1, \pm .447$.

[natheight=11.0007cm, natwidth=11.0007cm, height=11.1303cm, width=11.1303cm]
C:/sw50/temp/graphics/michmen$_{35.pdf}$
Designs for (linear approximation of) the Michaelis-Menten model $f\left(x;\boldsymbol{\theta}\right)=\theta_1 x/\left(\theta_2+x\right)$, $0\leq x\leq 10$ ($N=100$); assumed values $\theta_1=.2$, $\theta_2=.4$. D-optimal (Box-Lucas) design places half of the design points at each of .37 and 10.