# STOR 590:
# ADVANCED LINEAR MODELS
# Instructor: Richard L. Smith

## Class Notes:

## August 12, 2020

THE UNIVERSITY
of NORTH CAROLINA
at CHAPEL HILL

# BASICS OF LINEAR REGRESSION

$$y_i = x_{i0}\beta_0 + x_{i1}\beta_1 + ... + x_{ip}\beta_p + \epsilon_i, \ i = 1, ..., n$$

where $y_i$ is $i$th value of the observation of interest, $x_{i0}, ..., x_{ip}$ are the associated covariates, and $\epsilon_1, ..., \epsilon_n$ are random errors. Here $\beta_0, ..., \beta_p$ are the unknown parameters, or regression coefficients. Usually we assume $x_{i0} = 1$ and in that case we call $\beta_0$ the intercept. Matrix form:

$$y = X\beta + \epsilon.$$

Principle of least squares: Find $\beta_0, ..., \beta_p$ to minimize

$$L = \sum_i \left( y_i - \sum_j x_{ij}\beta_j \right)^2 .$$

Solve by calculus.

$$\begin{aligned}
\frac{\partial L}{\partial \beta_0} &= -2\sum_i \left( y_i - \sum_j x_{ij}\beta_j \right) x_{i0}, \\
\frac{\partial L}{\partial \beta_1} &= -2\sum_i \left( y_i - \sum_j x_{ij}\beta_j \right) x_{i1}, \\
&\cdots \\
\frac{\partial L}{\partial \beta_p} &= -2\sum_i \left( y_i - \sum_j x_{ij}\beta_j \right) x_{ip}.
\end{aligned}$$

We find the minimizing $\widehat{\beta}_0, ..., \widehat{\beta}_p$ by setting all the partial derivatives to 0, hence

$$\sum_i \left( y_i - \sum_j x_{ij}\widehat{\beta}_j \right) x_{ik} \;=\; 0,\;\; k = 0, ..., p.$$

Matrix notation:

$$X^T y - X^T X \widehat{\beta} \;=\; 0.$$

**The Normal Equations**

## Predicted values, $R^2$ and $R_a^2$

$$\widehat{y}_i = \sum_k x_{ik}\widehat{\beta}_k$$

or in matrix notation

$$\widehat{y} = X\widehat{\beta} = X(X^TX)^{-1}X^Ty.$$

We define (in case $x_{i0} \equiv 1$)

$$\begin{aligned} RSS &= \sum_i (\widehat{y}_i - y_i)^2, \\ TSS &= \sum_i (y_i - \bar{y})^2, \\ R^2 &= 1 - \frac{RSS}{TSS}. \end{aligned}$$

An alternative is the *adjusted* $R^2$ given by

$$R_a^2 = 1 - \frac{RSS/(n-p)}{TSS/(n-1)}.$$

## Summary Tables in R

The `summary` command in R produces a table of values that includes information about

1. *The residuals* — values $r_i = y_i - \widehat{y}_i$,

2. The standard errors, t-statistics and p-values of each of the parameter estimates.

For a parameter estimate $\widehat{\beta}_k$, R will give us a standard error $s_k$, then

$$t_k \; = \; \frac{\widehat{\beta}_k}{s_k}$$

is called the $k$th t statistic, so called because it has a $t_{n-p}$ distribution under the null hypothesis that $\beta_k = 0$.

## Confidence Interval for a Single Parameter

The confidence interval for $\beta_k$ at (two-sided) significance level $\alpha$ is

$$\widehat{\beta}_k \pm t_{n-p}^{\alpha/2} s_k$$

where $t_{n-p}^{\alpha/2}$ is the value exceeded with probability $\alpha/2$ by the $t_{n-p}$ distribution (in R: `qt(1-alpha/2,n-p)`).

## Confidence Interval for a Single Parameter

The confidence interval for $\widehat{\beta}_k$ at (two-sided) significance level $\alpha$ is

$$\widehat{\beta}_k \pm t_{n-p}^{\alpha/2} s_k$$

where $t_{n-p}^{\alpha/2}$ is the value exceeded with probability $\alpha/2$ by the $t_{n-p}$ distribution (in R: `qt(1-alpha/2,n-p)`).

# F Tests

Useful for testing whether a whole group of parameters is 0.

Suppose we have two models `lm1` and `lm2` where `lm1` is *nested* within `lm2` (in other words, every parameter that is in `lm2` is also in `lm1`, but not the other way round).

In the text, the two models are denoted by $\omega$ (`lm1`) and $\Omega$ (`lm2`). Suppose they have respectively $q$ and $p$ parameters, with $q < p$.

If model $\omega$ is true, then we have

$$F \;=\; \frac{(RSS_\omega - RSS_\Omega)/(p-q)}{RSS_\Omega/(n-p)} \;\sim\; F_{p-q,n-p}.$$

In R: `anova(lm1,lm2)`.

# INTRODUCING LEVERAGE, INFLUENCE AND COOK'S D STATISTIC

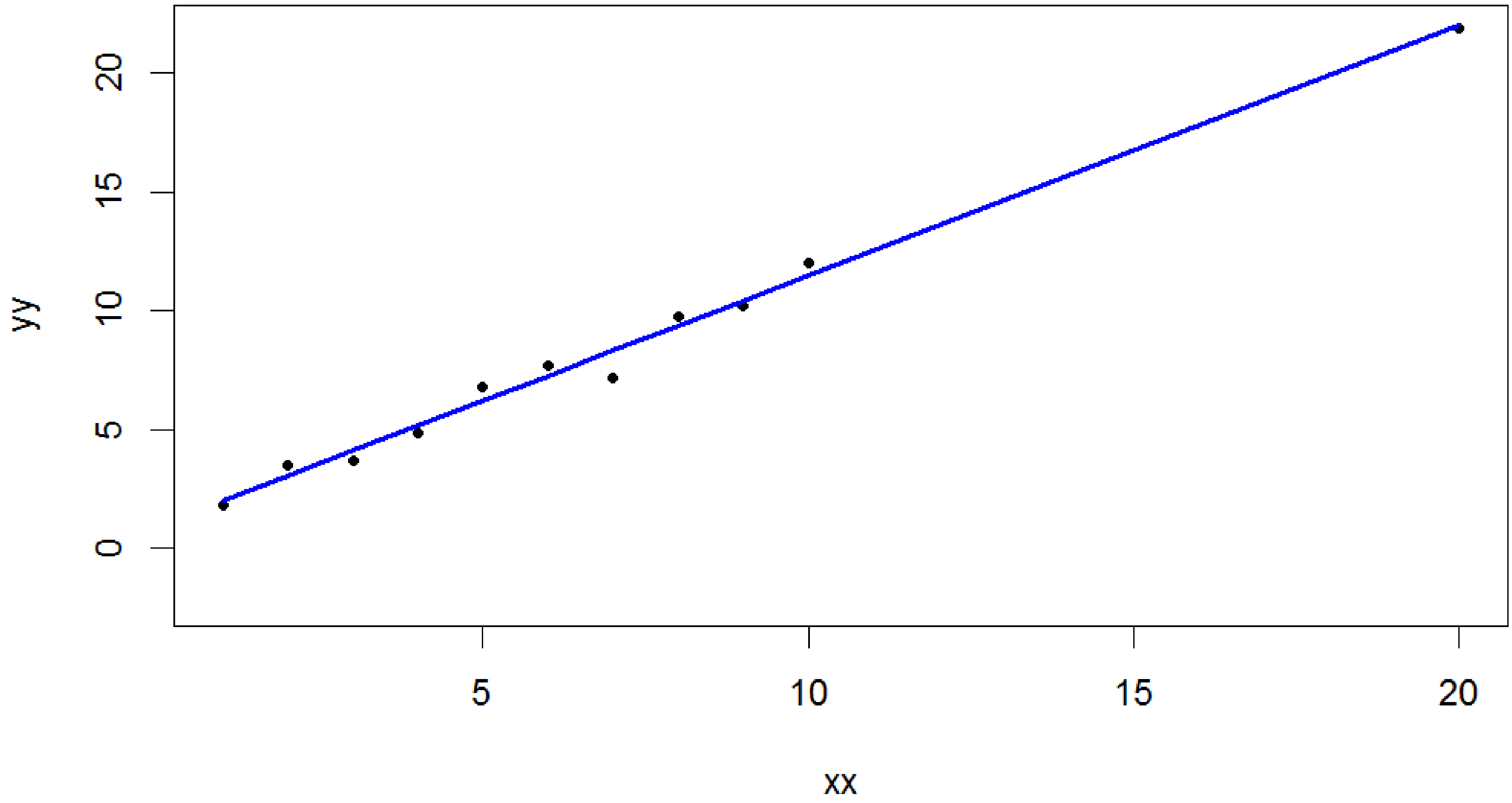
## Confidence and Prediction Intervals

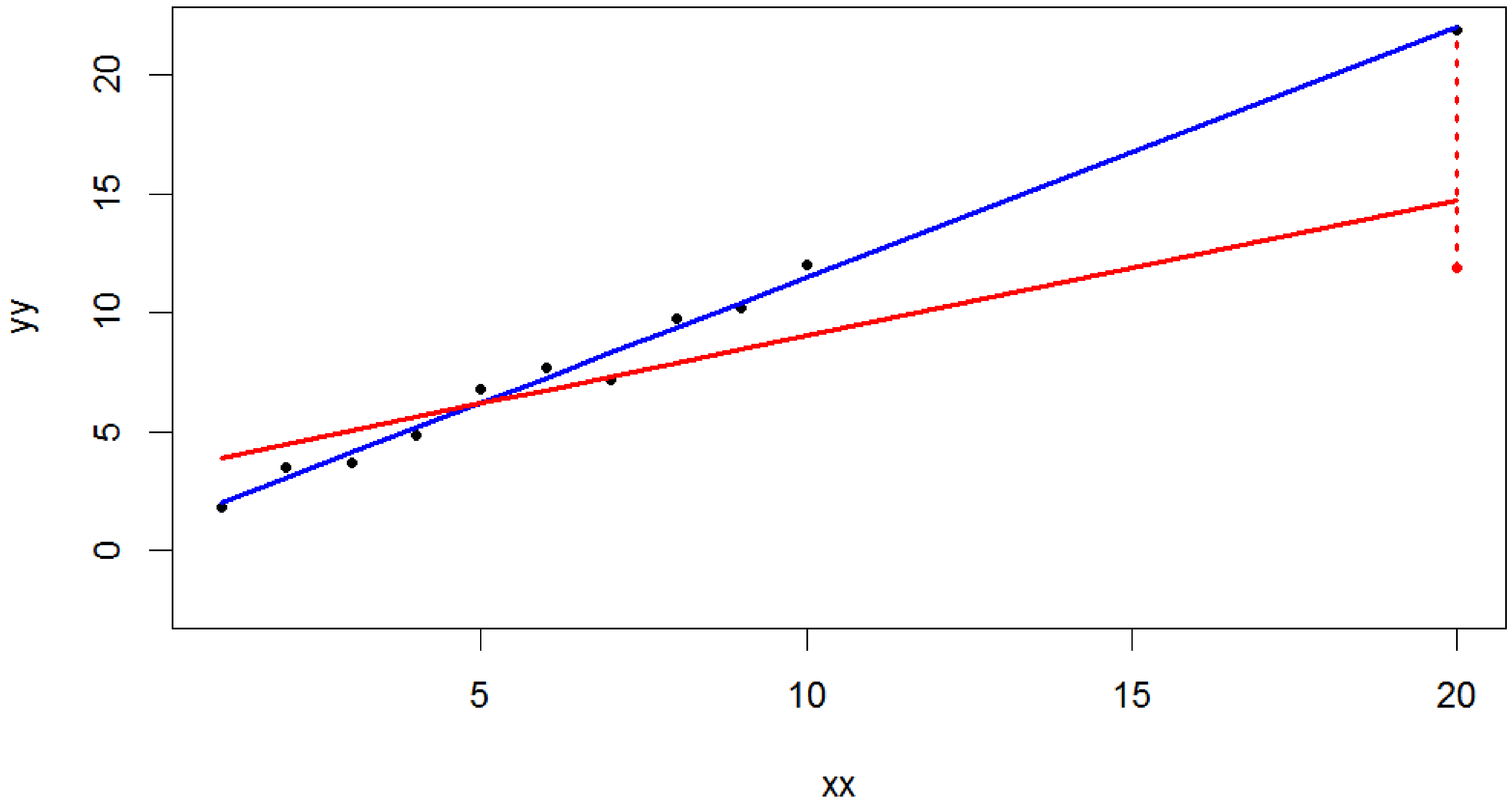
Consider a simple x-y plot with one “outlier” in the $x$ direction.

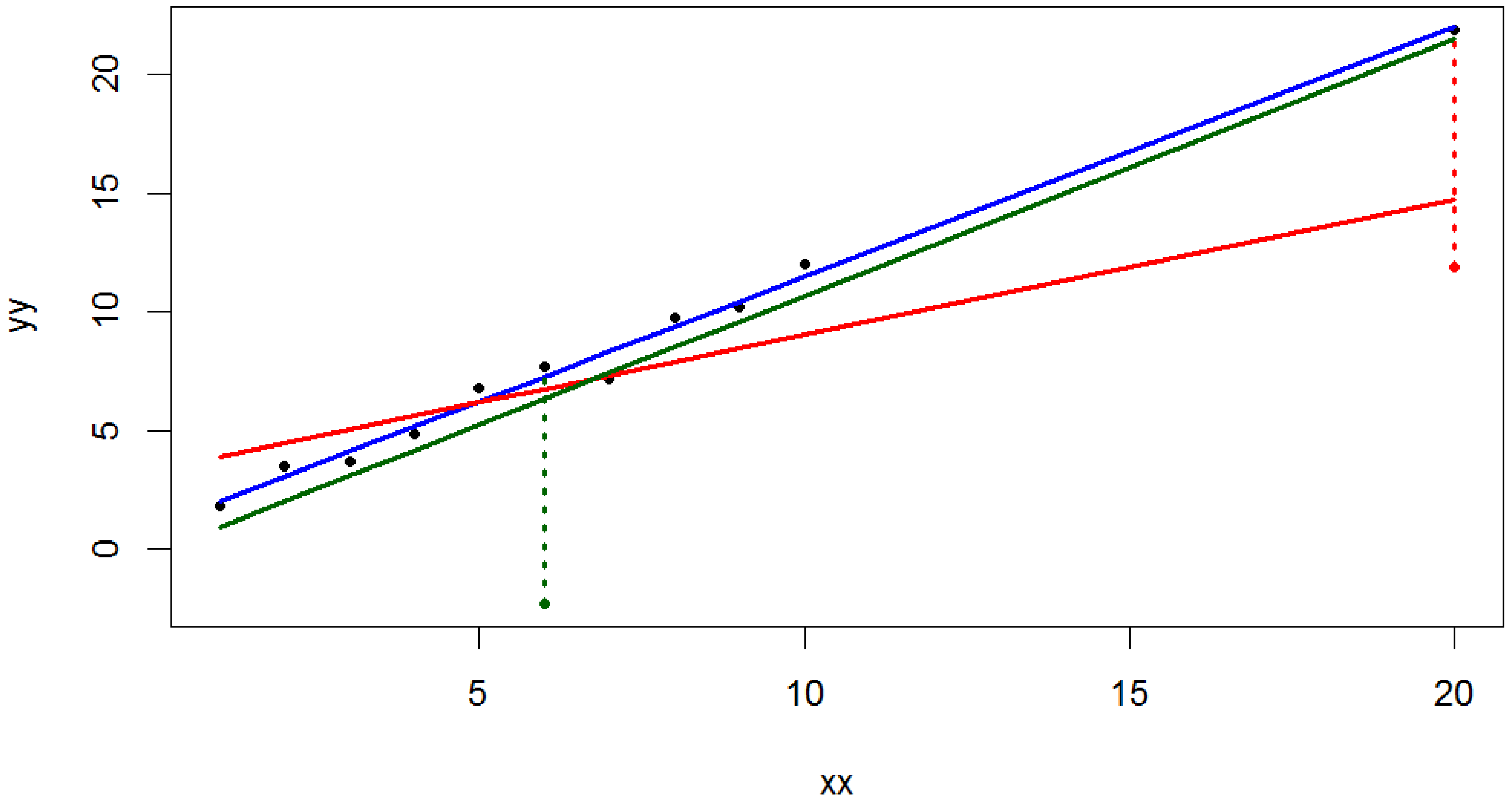
Consider the consequence of moving the corresponding $y$ value up or down.

The effect is much greater than if we took some arbitrary $x$ in the middle of the plot.

The difference is measured by *leverage*.


0
5
10
15
20
yy
5
10
15
20
xx


yy
0
5
10
15
20
5
10
15
20
xx


yy
0
5
10
15
20
5
10
15
20
xx

# THEORY

$$\begin{aligned}\widehat{y} &= X\widehat{\beta}\\ &= X(X^TX)^{-1}Xy\\ &= Hy\end{aligned}$$

where $H = X(X^TX)^{-1}X$ is known as the *hat matrix*.

$H$ is an $n \times n$ matrix whose *trace* (sum of diagonal entries) is $p+1$, the number of unknown parameters (including the intercept). The diagonal entries $h_i$, $i = 1, ..., n$ are called the *hatvalues*. "On average," the leverages are about $\frac{p+1}{n}$. Any point substantially larger than that is called a *point of high leverage*.

If you have previously fit a linear model to create an object "lmod", then

```
hatvalues(lmod)
```

will create the hat values.

# MY EXAMPLE

$$
\begin{aligned}
X &= \begin{pmatrix} 1 & 1 \\ 1 & 2 \\ \vdots & \vdots \\ 1 & 10 \\ 1 & 20 \end{pmatrix}, \; X^T X = \begin{pmatrix} 11 & 75 \\ 75 & 785 \end{pmatrix}, \\
(X^T X)^{-1} &= \frac{1}{3010} \begin{pmatrix} 785 & -75 \\ -75 & 11 \end{pmatrix}, \\
h_i &= 0.21,\; 0.18,\; 0.14,\; 0.12,\; 0.10,\; 0.09, \\
&\quad 0.09,\; 0.10,\; 0.11,\; 0.13,\; 0.73
\end{aligned}
$$

Observation 11 has about eight times the leverage of observation 6.

# R code for this example

```
X=matrix(c(rep(1,11),1:10,20),ncol=2)

# display X^T X
t(X) %*% X

library(’MASS’)

# display inverse of X^T X
ginv(t(X) %*% X)

# diagonal entries of the hat matrix, rounded to 2 decimal places
round(diag(X %*% (ginv(t(X) %*% X) %*% t(X))),2)
```

# Confidence and Prediction Intervals, 1

Properties of $H$:

1. H is *symmetric*

   Proof: $H^T = \{X(X^TX)^{-1}X^T\}^T = (X^T)^T(X^TX)^{-1}X^T = H$.

2. H is *idempotent* $(H^2 = H)$

   Proof: $H^2 = X(X^TX)^{-1}X^TX(X^TX)^{-1}X^T = X(X^TX)^{-1}X^T$.

3. $HX = X$

   Proof: $\{X(X^TX)^{-1}X^T\}X = XI = X$.

# Confidence and Prediction Intervals, 2

Properties of $\widehat{y}$:

1. $\widehat{y} = Hy = H(X\beta + \epsilon) = X\beta + H\epsilon$. Mean is $X\beta$.
2. The covariance matrix of $\widehat{y}$ is
$$\begin{aligned} E\left\{(\widehat{y} - X\beta)(\widehat{y} - X\beta)^T\right\} &= E\left\{H\epsilon(H\epsilon)^T\right\} \\ &= H \cdot E\left\{\epsilon\epsilon^T\right\} \cdot H^T \\ &= H \cdot \sigma^2 I \cdot H^T \\ &= \sigma^2 H. \end{aligned}$$
3. In particular, $Var(\widehat{y}_i) = h_i \sigma^2$.
4. For the standard linear model setup, $\sigma$ is estimated by the residual standard deviation $s$, for which $\frac{s^2}{\sigma^2} \sim \frac{\chi^2_{n-p}}{n-p}$ *independently of* $\widehat{\beta}$. Here $n$ is the number of observations and $p$ the number of covariates (including the intercept).

# Confidence and Prediction Intervals, 3

Suppose we want a 100$(1-\alpha)$% confidence interval for $x_i^T\beta$, $x_i$ the $i$'th column of $X$. We have that $\widehat{y}_i$ is an unbiased estimaor with variance $h_i\sigma^2$. Therefore:

$$\begin{aligned} \frac{\widehat{y}_i - x_i^T\beta}{\sqrt{h_i}\sigma} &\sim N[0,1], \\ \frac{\widehat{y}_i - x_i^T\beta}{\sqrt{h_i}s} &\sim t_{n-p} \end{aligned}$$

where $n$ is the number of observations and $p$ the number of parameters (including intercept). Therefore the desired *confidence interval* is

$$\widehat{y}_i \pm t_{n-p,1-\alpha/2} \cdot \sqrt{h_i} \cdot s.$$

where $t_{n-p,1-\alpha/2}$ is the $1-\alpha/2$ probability point of the $t_{n-p}$ distribution (In R: `qt(1-alpha/2,n-p)`).

# Confidence and Prediction Intervals, 4

Suppose, however, what we are really interested in is a *new* observation at $x_i$, say $y^* = x_i\beta + \epsilon^*$ where $\epsilon^* \sim N[0, \sigma^2]$ to mimic the errors in the regression already fitted. In that case,

$$y^* - \widehat{y}_i \;\sim\; N[0, \sigma^2(h_i + 1)]$$

were the $+1$ in the variance term reflects the variance of $\epsilon^*$.

So the *prediction standard error* is $s\sqrt{1 + h_i}$ and not $s\sqrt{h_i}$. The $100(1-\alpha)\%$ *prediction interval* for $y^*$ is

$$y^* \pm t_{n-p,1-\alpha/2} \cdot \sqrt{1 + h_i} \cdot s. \tag{1}$$

In R, you can do this one of two ways: either explicitly evaluate formula (1) using `summary(lmod)$sigma` for $s$ and `hatvalues(lmod)` for the vector of $h_i$, or use

`predict(lmod,interval='prediction',level=1-alpha)`.

The latter is usually easier to remember and use!

# Relevance to the Missing Votes Problem

1. To estimate the PNR in Bladen or Robeson county, we absolutely must take into account the natural varability of PNR, as well as the regression component.

2. The `se.fit` option with the `predict` command computes the confidence interval SE, not the prediction interval SE.

3. Therefore, we must either multiply the vector of confidence interval SEs by `sqrt((1+hatvalues(lmod))/hatvalues(lmod))` or (simpler) use the `interval='prediction'` option to compute prediction intervals directly.

4. This comment applies both to the original formulation of the question (estimate the probability of the observed value in Bladen and Robeson county) and the revised formulation (esitmate number of lost votes), but the latter is simpler (and more meaningful) because it works with the prediction intervals directly.

# COOK'S D STATISTIC

The most influential observations are those that have both large residuals and high leverage.

Cook's D statistic combines them both into a single measure.

Define $p$ as the number of regressors (including intercept), $\widehat{y}$ the vector of predicted values, $\widehat{y}_{(i)}$ the vector of predicted values when the $i$'th observation is omitted, and $\widehat{\sigma}^2$ the estimated residual variance.

$D_i$, the influence of observation $i$, is defined equivalently by

$$D_i = \frac{(\widehat{y}-\widehat{y}_{(i)})^T(\widehat{y}-\widehat{y}_{(i)})}{p\widehat{\sigma}^2} = \frac{1}{p}\cdot r_i^2 \cdot \frac{h_i}{1-h_i}.$$

Usually a $D_i$ close to 1 is considered meaningful — in other words, we should investigate whether that observation really does need to be corrected (or omitted).

# Comments on the rest of Chapter 1 (page 17 onwards)

1. *Robust Regression* (fit through the R package MASS) — this method became very popular for a while in the 1980s, but is less widely used now. You should be aware of it, but no need to study in depth.

2. *Weighted Least Squares.* Another method of accounting for heteroscedasticity is to weight each observation proportional to the sample size for that observation (in our example, the AbsBal variable). Faraway recommends *against* this option because the variances don't follow a simple scaling formula in practice. I suspect the same caveat applies with our example.

3. *Transformations of y.* Also called Box-Cox transformation. Could consider replacing PNR by logarithm or square root of PNR. One test: does this improve $R^2$?

4. *Transformations of x variables.* This could be a good idea if it improves the overall $R^2$ (or, equivalently, reduces the RSS). Faraway gives several examples. (Another variant would be to include interactions, e.g. cross-products of existing x variables. I know some of you tried that with our voting example. The criterion is whether the new variables improve the fit to a statistically significant extent.)

5. *Variable selection methods.* Several possibilities, e.g.
   (a) Maximize adjusted $R^2$ (simplest but not necessarily best)
   (b) Minimize AIC (or BIC, DIC,....)
   (c) Forward, backward or stepwise regression (numerous variants)
   (d) Newer "machine learning" methods, e.g. *lasso*

   None of these methods is universally "best" — choice is partly a matter of personal preference (and the size of the dataset)