STAT 525

Chapter 11
Remedial Measures for Regression

Dr. Qifan Song

Unequal Error Variances

- Consider $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$ where $\boldsymbol{\sigma}^2(\boldsymbol{\varepsilon}) = \mathbf{W}^{-1}$
  - Potentially correlated errors and unequal variances
- Special case: $\mathbf{W} = diag\{w_1, w_2, \cdots, w_n\}$
  - Heterogeneous variance or *heteroscedasticity*
  - Homogeneous variance or *homoscedasticity* if $w_1 = w_2 = \cdots = w_n = 1/\sigma^2$
  - Least square estimation still yields unbiased estimation, but is no longer optimal, and gives wrong uncertainty quantification
- Transformation of $\mathbf{X}$ or $\mathbf{Y}$ (e.g. Box-CoX) alone may unduly affect the relationship between $\mathbf{X}$ and $\mathbf{Y}$
- Error variance is often a function of $\mathbf{X}$ or $E[\mathbf{Y}]$

Transformation Approach

- Consider a transformation based on a known $\mathbf{W}$

$$\begin{aligned}
\mathbf{W}^{1/2}\mathbf{Y} &= \mathbf{W}^{1/2}\mathbf{X}\beta + \mathbf{W}^{1/2}\varepsilon \\
&\downarrow \\
\mathbf{Y}_w &= \mathbf{X}_w\boldsymbol{\beta} + \boldsymbol{\varepsilon}_w
\end{aligned}$$

- Can show $\mathrm{E}(\boldsymbol{\varepsilon}_w) = 0$ and $\boldsymbol{\sigma}^2(\boldsymbol{\varepsilon}_w) = \mathbf{I}$

- *Generalized least squares*: apply the least squares method to $\mathbf{Y}_w = \mathbf{X}_w\boldsymbol{\beta} + \boldsymbol{\varepsilon}_w$
  - It reduces to *weighted least squares* when $\mathbf{W}$ is a diagonal matrix
  - The transformation only requires that we know $\mathbf{W}$ up to some constant

Weighted Least Squares

- The least squares method minimizes

$$Q_w = (\mathbf{Y}_w - \mathbf{X}_w\boldsymbol{\beta})'(\mathbf{Y}_w - \mathbf{X}_w\boldsymbol{\beta}) = (\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})'\mathbf{W}(\mathbf{Y} - \mathbf{X}\boldsymbol{\beta})$$

  – When $\mathbf{W} = diag\{1/\sigma_1^2, 1/\sigma_2^2, \cdots, 1/\sigma_n^2\}$,

$$Q_w = \sum_{i=1}^{n} \frac{1}{\sigma_i^2}(Y_i - \mathbf{X}_i'\boldsymbol{\beta})^2$$

- By taking a derivative of Q_w, obtain normal equations:

$$(\mathbf{X}_w'\mathbf{X}_w)\mathbf{b} = \mathbf{X}_w'\mathbf{Y}_w \quad \rightarrow \quad (\mathbf{X}'\mathbf{W}\mathbf{X})\mathbf{b} = \mathbf{X}'\mathbf{W}\mathbf{Y}$$

- Solution of the normal equations:

$$(\mathbf{X}_w'\mathbf{X}_w)^{-1}\mathbf{X}_w'\mathbf{Y}_w \quad \rightarrow \quad b = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{Y}$$

  – Can also be viewed as maximum likelihood estimator (MLE).

Weighted Least Squares (Continued)

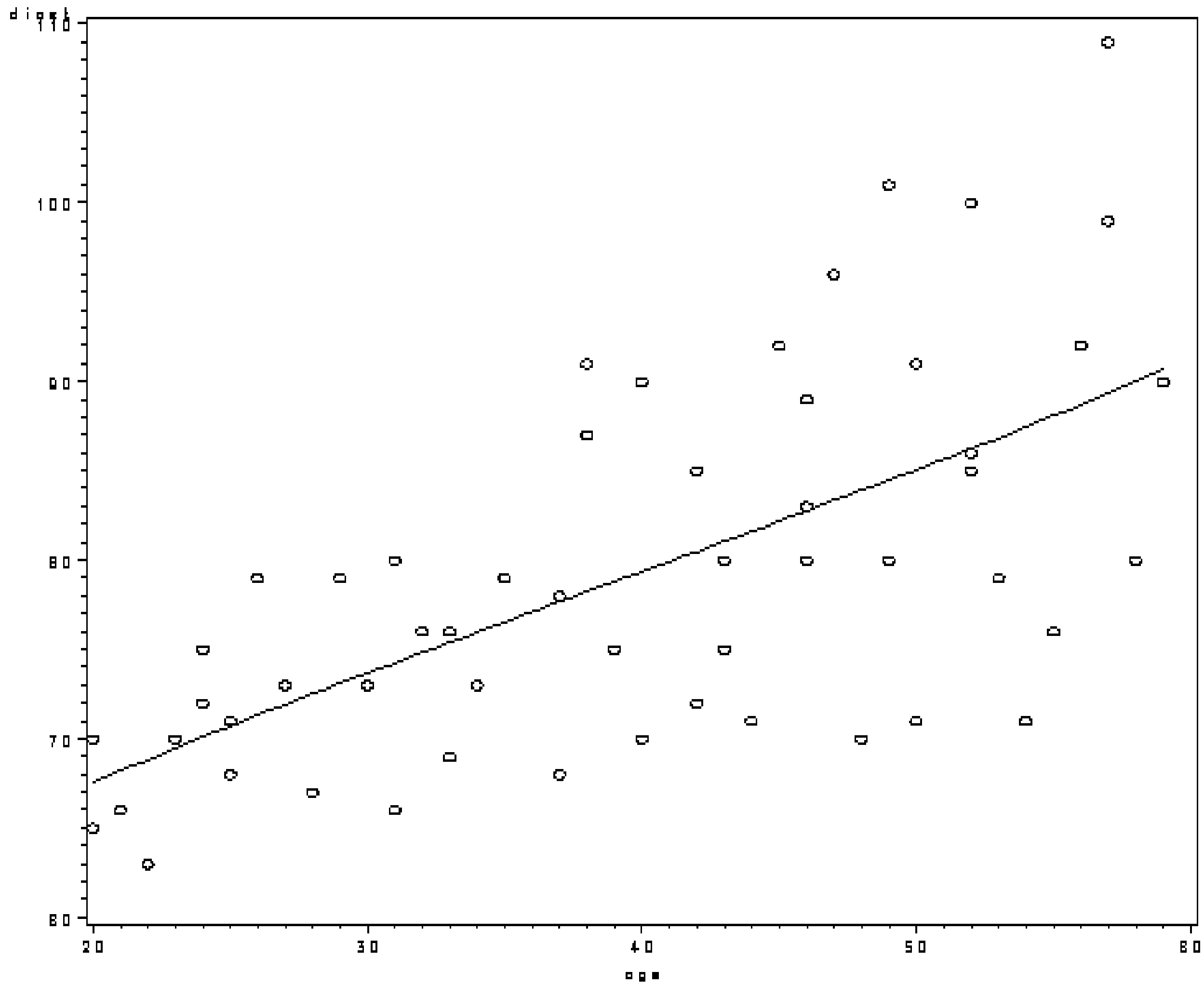
- Easy to do in SAS using the `weight` option
- Must determine optimal weights
- Optimal weights $\propto$ 1/variance
- Methods to determine weights, if no prior information of variance
  - Find relationship between the absolute residual and another variable and use this as a model for the standard deviation
  - Instead of the absolute residual, use the squared residual and find function for the variance
  - Use grouped data or approximately grouped data to estimate the variance

Example Page 427

- Interested in the relationship between diastolic blood pressure and age
- Have measurements on 54 adult women
- Age range is 20 to 60 years old
- Issue:
  - Variability increases as the mean increases
  - Appears to be nice linear relationship
  - Don't want to transform X or Y and lose this

```
data a1;
    infile 'U:\.www\datasets525\ch11ta01.txt';
    input age diast;
run; quit;
```

```
/* Scatter Plot */
proc sort data=a1; by age;
symbol v=circle i=sm70;
proc gplot data=a1;
    plot diast*age/frame;
run;
```



```
/* Fit a Regular Regression */
proc reg data=a1;
    model diast=age;
    output out=a2 r=resid;
run;
```

Analysis of Variance

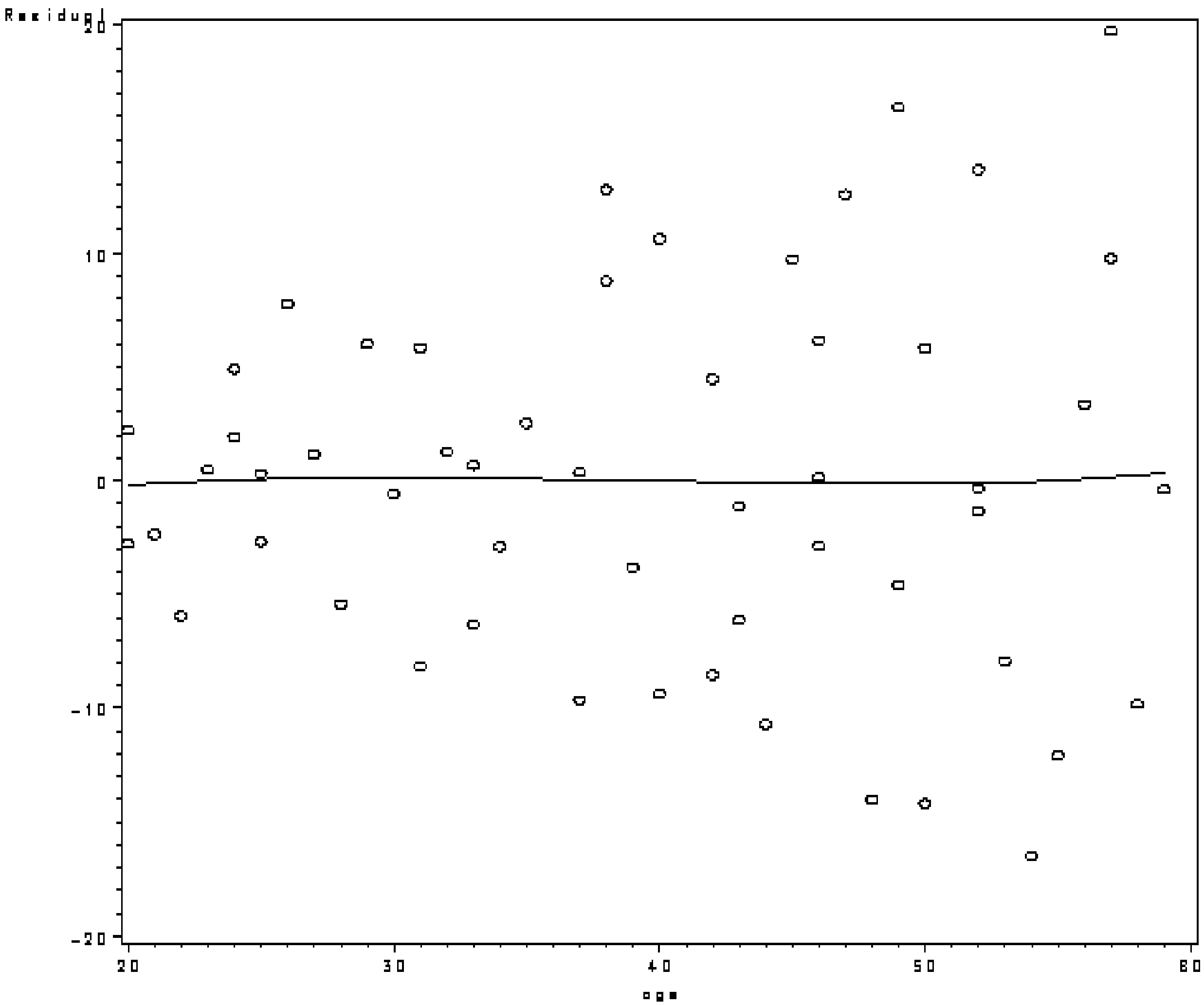
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	2374.96833	2374.96833	35.79	<.0001
Error	52	3450.36501	66.35317		
Corrected Total	53	5825.33333			

Root MSE	8.14575	R-Square	0.4077
Dependent Mean	79.11111	Adj R-Sq	0.3963
Coeff Var	10.29659		

Parameter Estimates

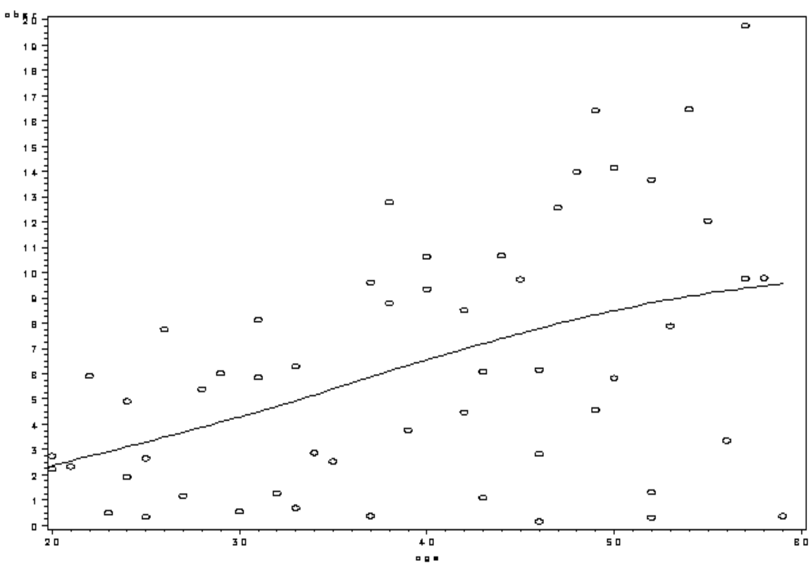
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	56.15693	3.99367	14.06	<.0001
age	1	0.58003	0.09695	5.98	<.0001

```
/* Residual Plot */
proc gplot data=a2;
    plot resid*age;
run; quit;
```

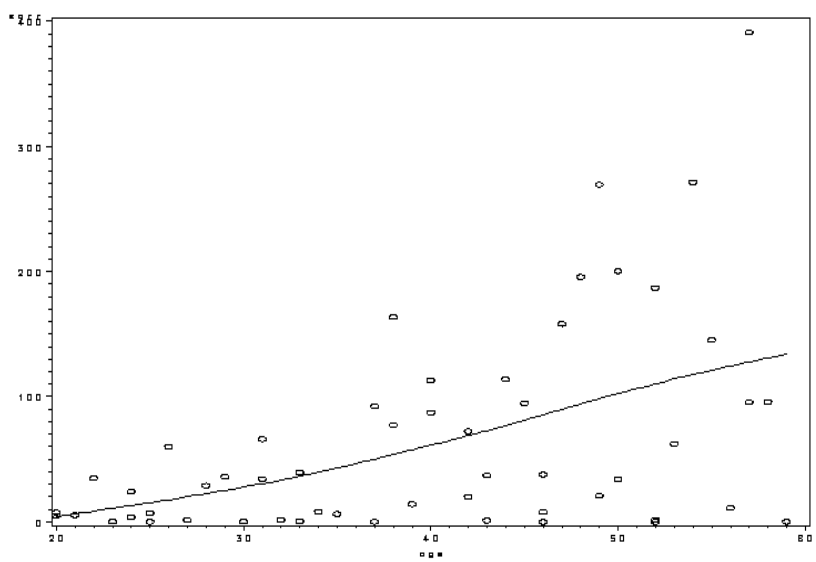


- The error variance increases as `age` increases

```
/* Find Pattern of Residuals vs Age */
data a2;
  set a2;
  absr=abs(resid);
  sqrr=resid*resid;

proc gplot data=a2;
   plot (resid absr sqrr)*age;
run;
```



abs(Residual) vs. Age



Residual^2 vs. Age

Construction of Weights

- Assume abs(res) is linearly related to age
- Fit least squares model and estimate σ_i

```
proc reg data=a2;
    model absr=age;
    output out=a3 p=shat;
run;
```

- Take Weight as $w_i = 1/\widehat{\sigma}_i^2$

```
data a3; set a3;
    wt=1/(shat*shat);

proc reg data=a3;
    model diast=age / clb;
    weight wt;
run; quit;
```

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	83.34082	83.34082	56.64	<.0001
Error	52	76.51351	1.47141		
Corrected Total	53	159.85432			

Root MSE	1.21302	R-Square	0.5214
Dependent Mean	73.55134	Adj R-Sq	0.5122
Coeff Var	1.64921		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	95% Confidence Limits	
Intercept	1	55.56577	2.52092	22.04	<.0001	50.50718	60.62436
age	1	0.59634	0.07924	7.53	<.0001	0.43734	0.75534

- Not much difference in the estimates but a slight reduction in the standard deviations. Should not interpret R^2 in this situation.

Ridge Regression as Multicollinearity Remedy

- Modification of least squares that overcomes multicollinearity problem
- Recall least squares suffers because $(\mathbf{X}'\mathbf{X})$ is almost singular thereby resulting in highly unstable parameter estimates
- Ridge regression results in biased but more stable estimates
- After standardizing data, we consider the correlation transformation so the normal equations are given by $\mathbf{r}_{XX}\mathbf{b} = \mathbf{r}_{YX}$. Since $\mathbf{r}_{XX}$ difficult to invert, we add a bias constant, c.

$$\mathbf{b}^R = (\mathbf{r}_{XX} + c\mathbf{I})^{-1}\mathbf{r}_{YX}$$

We then tranform it back to coefficient estimators for the orignal data.

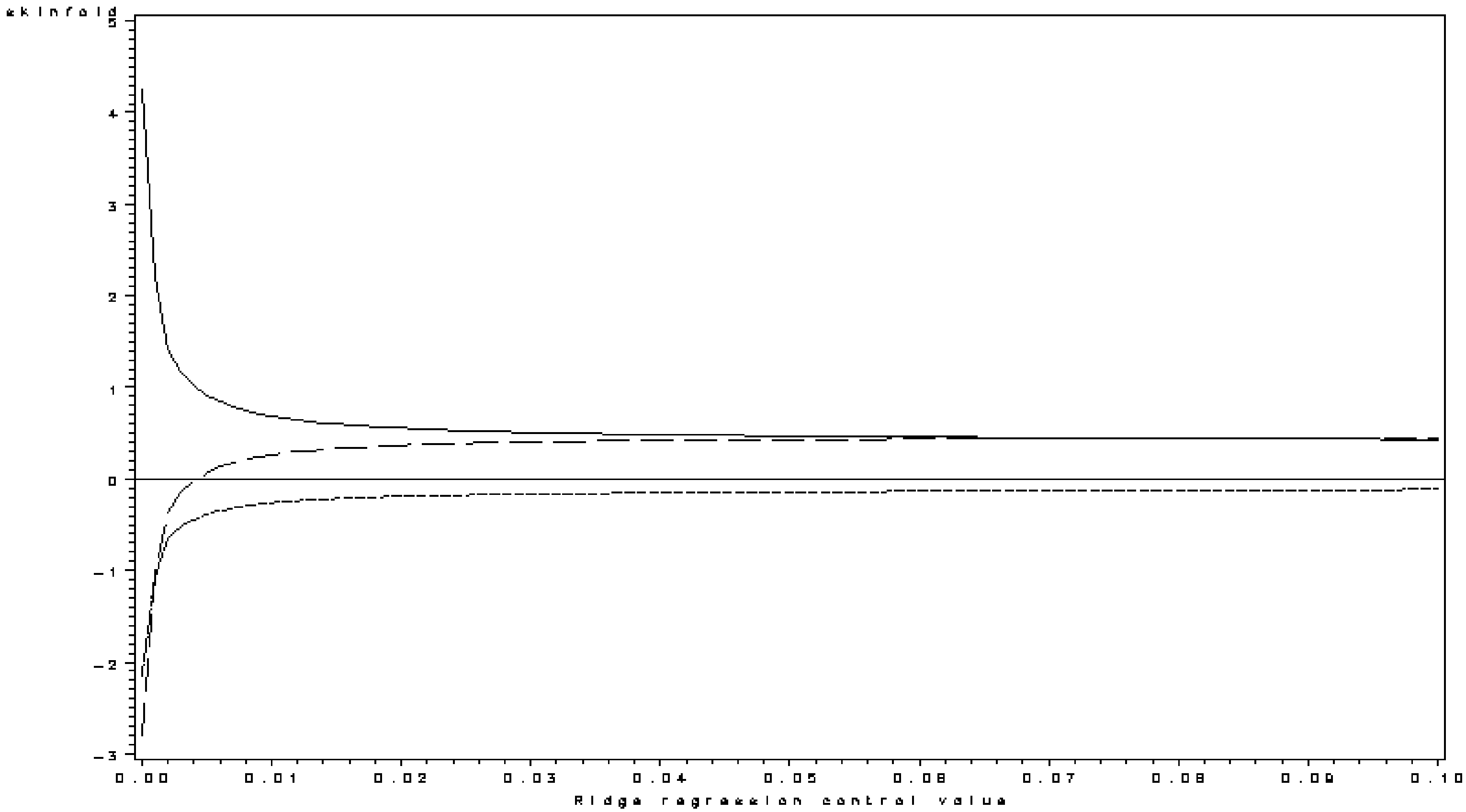
Choice of c

- Key to approach is choice of c
- Common to use the *ridge trace* and VIF's
  - Ridge trace: simultaneous plot of $p-1$ parameter estimates for different values of $c \geq 0$. Curves may fluctuate widely when c close to zero but eventually stabilize and slowly converge to 0.
  - VIF's tend to fall quickly as c moves away from zero and then change only moderately after that
- Choose c where things tend to "stabilize"
- `MODEL` statement of `PROC REG` has option `ridge=c`

SAS Commands

```
data a1;
    infile 'U:\.www\datasets525\ch07ta01.txt';
    input skinfold thigh midarm fat;

/* ridge estimation are stored in the dataset designated by outest option */
proc reg data=a1 outest=b;
    model fat=skinfold thigh midarm /ridge=0 to .1 by .001;
run;

symbol1 v='' i=sm5 l=1;
symbol2 v='' i=sm5 l=2;
symbol3 v='' i=sm5 l=3;
proc gplot;
    plot (skinfold thigh midarm)*_ridge_ / overlay vref=0;
run; quit;
```

Ridge Trace




```
/* Another Way to get the Ridge Trace Plot */
proc reg data=a1 outest=b;
    model fat=skinfold thigh midarm /ridge=0 to .1 by .001;
    plot / ridgeplot vref=0;
run;
```

Robust Regression with Influential Cases

- Want procedure that is not sensitive to outliers
- Focus on parameters which minimizes
  - sum of absolute values of residuals (LAR: Least Absolute Residuals)
  - median of the squares of residuals (LMS: Least Median of Squares)
- Could also consider iterating through weighted LS where the residual value is used to determine the weight (IRLS)
- See pages 439-449 for more details
- Both robust and ridge regression are limited by more difficult assessments of precision (i.e., standard errors). Bootstrapping is often used.

Iteratively Reweighted Least Squares Using PROC NLIN

- PROC NLIN allows to define weights as a function

```
/* NOHALV: removes the restriction that the objective value must
           decrease at every iteration */
PROC NLIN DATA=a1 NOHALV;
    PARMS b0=0 b1=0;
    MODEL diast = b0+b1*age;
    resid = diast-MODEL.diast;
    _WEIGHT_ = 1/(resid**2);
RUN; QUIT;
```

The NLIN Procedure

Source	DF	Sum of Squares	Mean Square	F Value	Approx Pr > F
Model	1	1.877E23	1.877E23	1.81E23	<.0001
Error	52	54.0000	1.0385		
Corrected Total	53	1.877E23			

Parameter	Estimate	Approx Std Error	Approximate 95% Confidence Limits	
b0	56.8462	3.8E-11	56.8462	56.8462
b1	0.5385	1.27E-12	0.5385	0.5385

Approximate Correlation Matrix

	b0	b1
b0	1.0000000	-0.9999964
b1	-0.9999964	1.0000000

Nonparametric Regression

- Helpful in exploring the nature of the response function
- i=sm## is one such approach (Spline)
- All version have some sort of smoothing, via local averaging or global smooth basis functions
- See pages 449-453 for more details
- Interesting theory but confidence intervals and significant tests not fully developed

Machine Learning algorithms

Flexible modeling with few inference tools. Usually requires iterative optimization algorithms.

Regression Trees

- Piecewise constant regression function
- Basically partition the X space into rectangles
- Predicted value is mean of responses in rectangle
- Minimize SSE via greedy search (sequentially partitioning)
- Trade off between minimizing SSE and complexity

Neural Networks

- $y = B_1 \circ \sigma \circ B_2 \circ \sigma \ldots B_m \circ x$
- Combination of nonlinear activation function σ and linear mapping B_i's.
- Estimate B_i's via minimizing squared loss
- Highly non-convex optimization task
- Foundation of deep learning

Evaluating Precision in Nonstandard Situations

- Standard methods for evaluating the precision of sample estimates may not be available or may only be approximately applicable when the sample size is large
  - Ridge regression
  - Robust regression
- **Bootstrapping** provides estimates of the precision of sample estimates
  - **Very** important theoretical development that has had a major impact on applied statistics
  - Resampling idea: use the sample to generate a "population" and generate new "samples" from such "population"
  - Use the pool of estimates from new "samples" to profile sample estimates (i.e., parameters of "population")

Resampling Residuals (Fixed X sampling)

- Take the residuals $\{e_1, e_2, \cdots, e_n\}$ as the "population" of error term ϵ
  - Sample ϵ_i^* from $\{e_1, e_2, \cdots, e_n\}$
  - Let $Y_i^* = b_0 + b_1 X_i + \epsilon_i^*$
  - In the new "sample", the i-th observation is (X_i, Y_i^*)
  - Assume constant error variances
- Useful when
  - errors have unknown distribution (but constant variance), and/or
  - want to preserve predictors
- Examples of use:
  - Ridge regression
- May sample ϵ^* from a "parametric population" of residuals

Resampling Pairs (Random X Sampling)

- Useful when
  - Doubt about the adequacy of the regression function being fitted
  - Unequal error variances
  - Predictor variables cannot be regarded as fixed
- Take $\{(X_i, Y_i) : i = 1, 2, \cdots, n\}$ as the "population" of (X, Y)
  - For the new "sample", the i-th observation (X_i^*, Y_i^*) is sampled from the "population" $\{(X_i, Y_i) : i = 1, 2, \cdots, n\}$
- Examples of use:
  - Weighted regression

Bootstrap Inference

- A total of B new "samples" can be generated, with each new "sample" providing an estimate of the parameter, say $b_1^{(k)}$ for β_1 from k-th new "sample"
  - Use $\{b_1^{(k)} : k = 1, 2, \cdots, B\}$ to understand the population property of b_1

- Bias

$$Bias = E\{b_1\} - \beta_1 \Longrightarrow \widehat{\text{Bias}}_{boot} = \bar{b}_1^* - b_1$$

where $\bar{b}_1^* = \sum_{k=1}^{B} b_1^{(k)}/B$

- Variance

$$Var = E\{(b_1 - E\{b_1\})^2\} \Longrightarrow \widehat{\text{Var}}_{boot} = \frac{1}{B}\sum_{k=1}^{B}(b_1^{(k)} - \bar{b}_1^*)^2$$

Bootstrap Confidence intervals

- CI for β_1 with unbiased estimator b_1

$$(b_1^*(\alpha/2), b_1^*(1-\alpha/2))$$

  - $b_1^*(\alpha/2)$ is the $(\alpha/2)\times 100$ percentile of $\{b_1^{(k)} : k = 1, 2, \cdots, B\}$
  - $b_1^*(1-\alpha/2)$ is the $(1-\alpha/2) \times 100$ percentile of $\{b_1^{(k)} : k = 1, 2, \cdots, B\}$

- *Reflection Method*: CI for β_1 with biased estimator b_1

$$(b_1 - d_2,\ b_1 + d_1)$$

  - $d_1 = b_1 - b_1^*(\alpha/2)$
  - $d_2 = b_1^*(1-\alpha/2) - b_1$

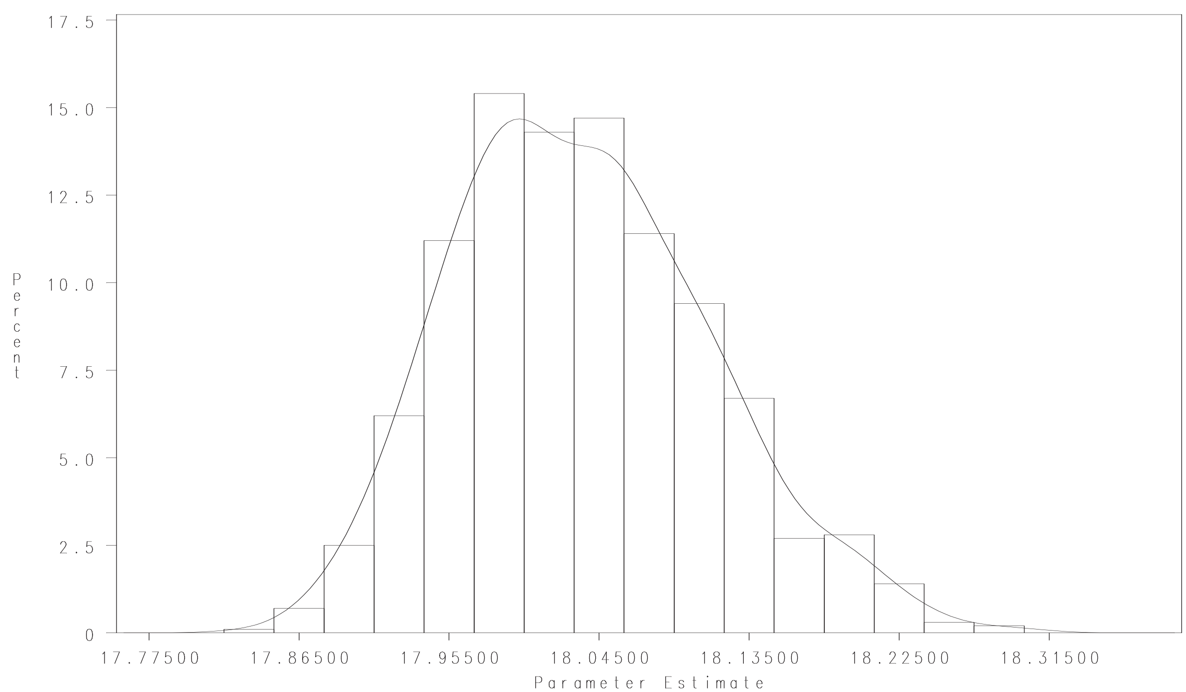
Example: Typographical Errors (4.12 on Page 173)

```
options nocenter; goptions colors=(none);

/* ----Read in initial data set and fit the model----*/
data a1;
    infile 'U:\.www\datasets525\CH04PR12.txt';
    input y x;

proc reg;
    model y=x / noint clb;
    output out=a2 p=pred r=res;
run;
---------------------------------------------------------------------
Output from Proc Reg

                      Parameter       Standard
Variable     DF        Estimate          Error    t Value    Pr > |t|
x             1        18.02830        0.07948     226.82      <.0001

Variable     DF        95% Confidence Limits
x             1        17.85336       18.20325
---------------------------------------------------------------------
```

```
/* Resample Residuals */

/* Create a data set that contains 1000 copies of the predictor
   variable and associated fitted value from the regression */
data pred; set a2;
    do sample=1 to 1000;
        output;
        keep sample x pred;
    end;
proc sort; by sample;
run;

/* Randomly sample (with replacement) the residuals => 1000 copies */
/* PROC SURVEYSELECT: Selecting random samples */
/* METHOD=URS: Select with equal probability & with replacement */
/* SAMPSIZE: Specifies the sample size */
/* REP: Number of samples (i.e., datasets) */
/* OUTHITS: Includes a separate observation in the output dataset for each
            selection when the same unit is selected more than once */
/* ID: variables to be included in the output dataset, all by default */
proc surveyselect data=a2 method=urs sampsize=12 rep=1000 outhits out=res;
   id res;
run;
```

```
/* Merge the fitted values and the residuals, and generate new y */
data new;
    merge pred res;
    ynew = pred + res;
run;

/* Perform regression on each new sample and store parameter estimate
   results in a dataset called parm */
/* The ods listing turns off the output going into the output window */
ods listing close;
proc reg data=new;
    model ynew=x / noint;
    by sample;
ods output ParameterEstimates=parm;
ods listing;

/* Generate histogram and approximate the density */
/* PCTLPRE: Specifies prefixes to create variables names for PCTLPTS */
/* PCTLPTS: Specifies percentiles to compute */
proc univariate noprint data=parm;
    var Estimate;
    histogram Estimate / kernel ;
    output out=a4 mean=bmean std=bsterr pctlpre=perc_ pctlpts=2.5,5,95,97.5;

proc print data=a4; run; quit;
```

```
--------------------------------------------------------------------------------
Results from Bootstrapping

Obs   bmean    bsterr  perc_2_5   perc_5  perc_95  perc_97_5
 1  18.0348  0.076574   17.9038  17.9200  18.1747    18.1986

Bias = 18.0348-18.0283 = 0.0065 (quite small)

Percentile :   (17.9038, 18.1986)
Reflection :   (17.8580, 18.1475)
--------------------------------------------------------------------------------
```



Histogram of $\{b_1^{(k)} : k = 1, 2, \cdots, 1000\}$

- The way to resample in the example is the easiest to implement
- But it is not a computationally efficient way to do resampling

Chapter Review

- Weighted least squares for unequal error variances
- Ridge regression for multicollinearity problem
- Robust regression for outliers / influential points
- Regression tree for nonparametric regression
- Evaluating precision in nonstandard situations using bootstrapping