

Statistics 512: Applied Linear Models

Topic 2

Topic Overview

This topic will cover

- Regression Diagnostics
- Remedial Measures
- Some other Miscellaneous Topics

Chapter 3: Diagnostics and Remedial Measures

Diagnostics – Look at the data to diagnose situations where the assumptions of our model are violated.

Some diagnostics check the *assumptions* of our model. Other diagnostics check the *influence* of different data points.

Remedies – Changes in analytic strategy to fix these problems.

What do we need to check??

- Main Assumptions: Errors are independent, normal random variables with common variance σ^2 .
- Are there “outlying” values for the predictor variables that could unduly influence the regression model?
- Is the model appropriate? Does the assumption of linearity make sense?
- Are there outliers? (Generally, the term *outlier* refers to a response that is vastly different from the other responses – see KNNL, page 108.)

How to get started...

1. Look at the data.
2. **Look at the data.**

3. **LOOK AT THE DATA.**

Before trying to describe the relationship between a response variable (Y) and an explanatory variable (X), we should look at the distributions of these variables. We should always look at X as well as Y , since if Y depends on X , looking at Y alone may not be very informative.

Section 3.1: Diagnostics for X

We do not make any specific assumptions about X . However, understanding what is going on with X is necessary to interpreting what is going on with Y . So we look at some basic summaries of the X variables to get oriented. However, we are not checking our assumptions at this point.

- If X has many values, use `proc univariate`.
- If X has only a few values, use `proc freq` or the `freq` option in `proc univariate`.
- Examine the distribution of X . Is it skewed? Are there outliers? Important statistics to consider include mean, standard deviation, median, mode, and range.
- *Box plots* and *Stem-and-leaf plots* are useful.
- Do the values of X depend on time (order in which the data were collected)? (*Sequence plots*)

Example (Toluca Company, page 100)

See program `knnl101.sas` for the code used to execute `proc univariate`.

```
data toluca;
  infile 'h:\System\Desktop\CH01TA01.DAT';
  input lotsize workhrs;
  seq=_n_;
proc print data=toluca;
```

Obs	lotsize	workhrs	seq
1	80	399	1
2	30	121	2
3	50	221	3
4	90	376	4
5	70	361	5
.	.	.	.
.	.	.	.

```
proc univariate data=toluca plot;
  var lotsize workhrs;
```

Moments			
N	25	Sum Weights	25
Mean	70	Sum Observations	1750
Std Deviation	28.7228132	Variance	825
Skewness	-0.1032081	Kurtosis	-1.0794107
Uncorrected SS	142300	Corrected SS	19800
Coeff Variation	41.0325903	Std Error Mean	5.74456265

Basic Statistical Measures			
Location		Variability	
Mean	70.00000	Std Deviation	28.72281
Median	70.00000	Variance	825.00000
Mode	90.00000	Range	100.00000
		Interquartile Range	40.00000

Tests for Location: $\mu_0=0$

Test	-Statistic-	-----p Value-----
Student's t	t 12.18544	Pr > t <.0001
Sign	M 12.5	Pr >= M <.0001
Signed Rank	S 162.5	Pr >= S <.0001

Quantiles (Definition 5)

Quantile	Estimate
100% Max	120
99%	120
95%	110
90%	110
75% Q3	90
50% Median	70
25% Q1	50
10%	30
5%	30
1%	20
0% Min	20

Extreme Observations

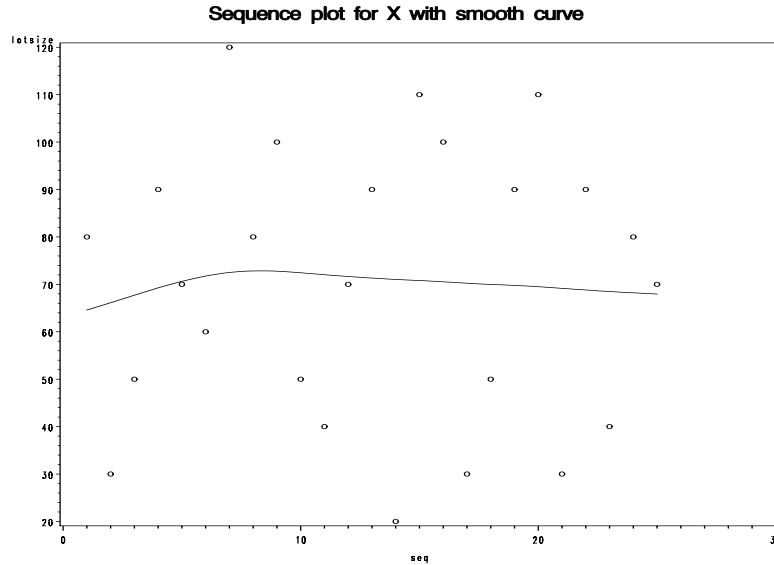
----Lowest----		----Highest----	
Value	Obs	Value	Obs
20	14	100	9
30	21	100	16
30	17	110	15
30	2	110	20
40	23	120	7

Stem Leaf	#	Boxplot
12 0	1	
11 00	2	
10 00	2	
9 0000	4	+-----+
8 000	3	
7 000	3	*---*---
6 0	1	
5 000	3	+-----+
4 00	2	
3 000	3	
2 0	1	
-----+		

```

title1 'Sequence plot for X with smooth curve';
symbol1 v=circle i=sm70;
proc gplot data=toluca;
    plot lotsize*seq;

```



Normal Distributions

Our model does *not* state that X comes from a single normal population.

Nor must Y come from a single normal population... (We assumed $Y|X$ and the residuals are normal, not Y itself).

In some case, X and/or Y may be normal, or their distributions may be unusual, and it can be useful to check this.

We DO assume that the residuals are normal, and we will also use the following technique to check this (later).

Normal Quantile Plots: The Basic Idea

Consider $n = 5$ observations iid from $N(0, 1)$

From Table B.1, we find

$$\begin{aligned}
 P(z \leq -0.84) &= 0.20 \\
 P(-0.84 < z \leq -0.25) &= 0.20 \\
 P(-0.25 < z \leq 0.25) &= 0.20 \\
 P(0.25 < z \leq 0.84) &= 0.20 \\
 P(z > 0.84) &= 0.20
 \end{aligned}$$

So we expect, *on average*,

One observation ≤ -0.84
 One observation in $(-0.84, -0.25)$
 One observation in $(-0.25, 0.25)$
 One observation in $(-0.25, 0.84)$
 One observation > 0.84

Paraphrasing KNNL, page 111, “Statistical theory has shown that for a normal random variable with mean 0 and variance 1, a good approximation of the expected value of the k th smallest observation in a random sample of size n is

$$Z_i = \Phi^{-1} \left(\frac{k - 0.375}{n + 0.25} \right), k = 1, \dots, n$$

where Φ^{-1} is the inverse of the standard normal cdf, i.e., the function that gives the normal percentiles. (I won’t go into the 0.375 or 0.25 here, but they were suggested by a statistician named Blom. Later in the semester, we will see how to get these values with `proc rank`. Also, there is an issue of how ties are handled. For now, let’s just let SAS give us the graph and not worry about the details.)

The Algorithm

- Plot the order statistics $X_{(i)}$ vs Z_i
- The standardized X variable is $Z = (X - \mu)/\sigma$
- So, $X = \mu + \sigma Z$
- If the data are approximately normal, the relationship will be approximately linear with slope close to σ and intercept close to μ .

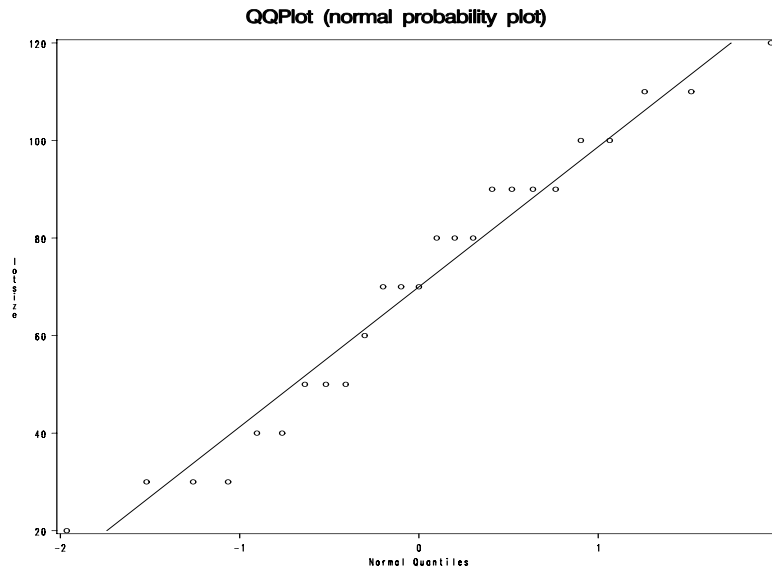
```

title1 'QQPlot (normal probability plot)';
proc univariate data=toluca noprint;
    qqplot lotsize workhrs / normal (L=1 mu=est sigma=est);

```

The options (after the `/`) tell SAS to also draw a straight line for comparison. The command `noprint` tells SAS NOT to print all the descriptive statistics, just make the qqplots.

Bottom line: To get a quantile plot, use `proc univariate` with a `qqplot` statement. If it looks roughly like a straight line, the variable approximately has a normal distribution. If it does not look straight, it is not normal.



Sections 3.2-3.3: Diagnostics for Residuals

Model: $Y_i = \beta_0 + \beta_1 X_i + \epsilon_i$,

where the ϵ_i are *independent, normal*, and have *constant variance*, that is $\epsilon_i \sim^{iid} N(0, \sigma^2)$.

The e_i should be similar to the ϵ_i ,

How do we check this?

PLOT

PLOT

PLOT

Questions Addressed by Diagnostics for Residuals

- Is the relationship linear?
- Does the variance depend on X ?
- Are the errors normal?
- Are there outliers?
- Are the errors dependent on order or each other?

These are addressed by examining various plots. See programs `knnl105.sas` and `knnl110.sas` for the residuals analysis.

Is the Relationship Linear?

Plot Y vs X .

Plot e vs X (residual plot)

Residual plot emphasizes deviations from linear pattern.

Example of a non-linear relationship

Make a dataset that we know is quadratic, not linear. Use the equation

$$Y = 30 - 10X + X^2 + N(0, 25^2)$$

```
data quad;
  do x = 1 to 30;
    y = x*x - 10*x + 30 + 25*normal(0);
    output;
  end;
proc reg data = quad;
  model y = x;
  output out = diagquad r = resid;
run;
```

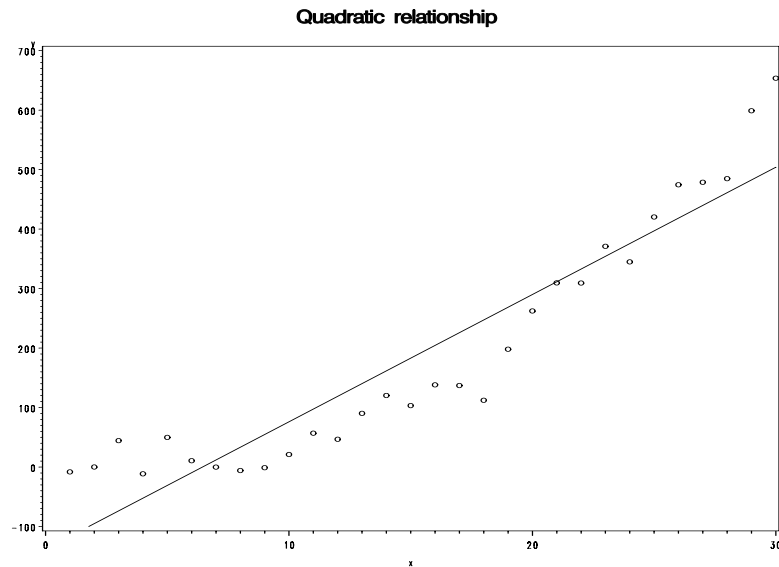
Analysis of Variance					
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	950768	950768	172.44	<.0001
Error	28	154382	5513.64182		
Corrected Total	29	1105150			

```
symbol1 v = circle i = rl;
title1 'Quadratic relationship';
proc gplot data = diagquad;
plot y*x;
```

The regression is significant, but from the plot, it is evident that the straight line does not fit the data very well. If we fail to notice this, we will make incorrect conclusions based on this regression.

Now draw it with a smooth curve instead of a straight line.

```
symbol1 v = circle i = sm60;
proc gplot data = diagquad;
  plot y*x;
```



Now plot the residuals vs X .

```
proc gplot data = diagquad;
    plot resid*x/ vref = 0;
```

The result is even more dramatic. This is clearly NOT a random scatter. Need to modify the model.

Does the variance depend on X ?

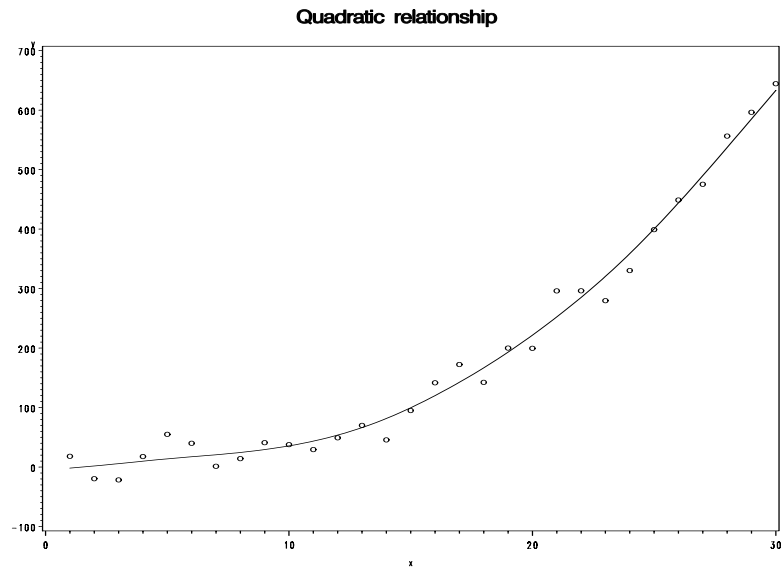
- Plot Y vs X
- Plot e vs X
- Plot of e vs X will emphasize problems with the variance assumption.
- If there is a problem with the constancy of variance assumption, you will see the differences in the vertical distance between points at similar X 's. (For example, the range near $X = 1$ might be $Y = 3$ to 7 while near $X = 10$ it might be $Y = 21$ to 41).

Example of non-constant variance (heteroscedastic)

Make a dataset whose variance changes with X .

$$Y = 30 + 100X + N(0, 100X^2)$$

(knnl104a.sas)



```

Data het;
  do x=1 to 100;
    y=100*x+30+10*x*normal(0);
    output;
  end;
run;

proc reg data=het;
  model y=x;
  output out=a3 r=resid;
run;
symbol1 v=circle i=sm60;
proc gplot data=a3;
  plot y*x/frame;

```

See how the scatter of the points away from the line increases with X .

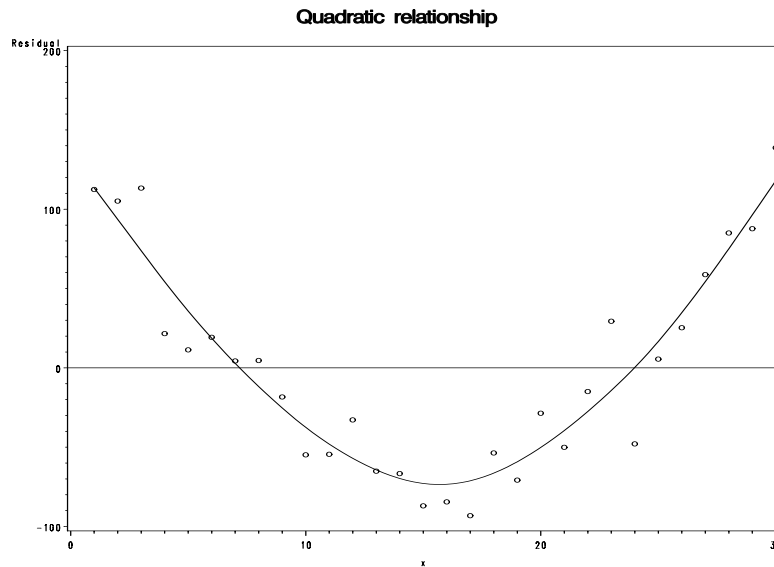
Now look at the residual plot. It is more obvious here.

```

proc gplot data=a3;
  plot resid*x/vref=0;

```

This has what we call a megaphone shape. The variance is clearly not constant here.



Are the errors normal?

The real question is whether the distribution of the errors is far enough away from normal to invalidate our confidence intervals and significance tests.

Look at the residuals' distribution.

Use a normal quantile plot (qqplot) and a histogram (knnl110.sas).

```
data toluca;
    infile 'H:\System\Desktop\CH01TA01.DAT';
    input lotsize workhrs;

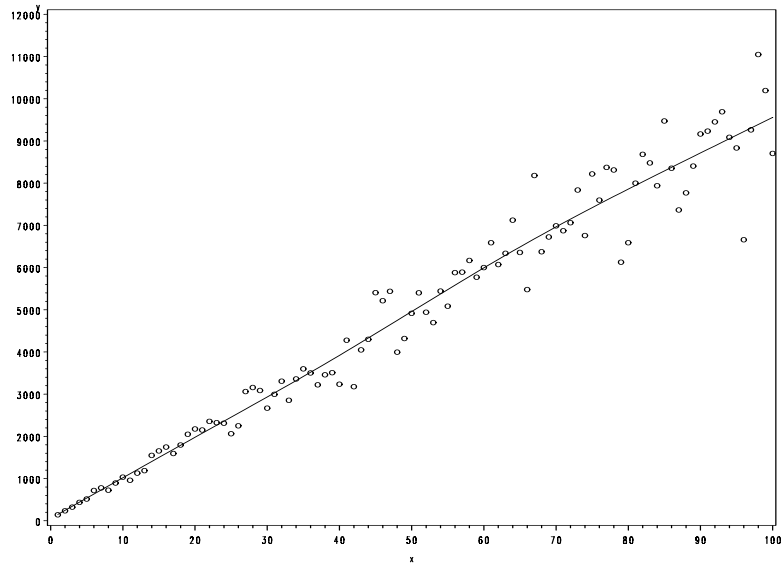
proc reg data=toluca;
    model workhrs=lotsize;
    output out=diag r=resid;

proc univariate data=diag plot normal;
    var resid;
    histogram resid / normal kernel(L=2);
    qqplot resid / normal (L=1 mu=est sigma=est);
```

The Toluca example looks pretty good.

Now let's see what happens when the normality assumption is violated.

```
data expo;
    do x = 1 to 100;
        y = 100*x+30+10*ranexp(1); output;
    end;
```



```
proc reg data = expo;
    model y = x; output out = diagexpo r = resid;
proc univariate data = diagexpo plot normal;
    var resid;
    histogram resid / normal kernel (L = 2);
    qqplot resid / normal (L = 1 mu = est sigma = est);
```

Tests for Normality

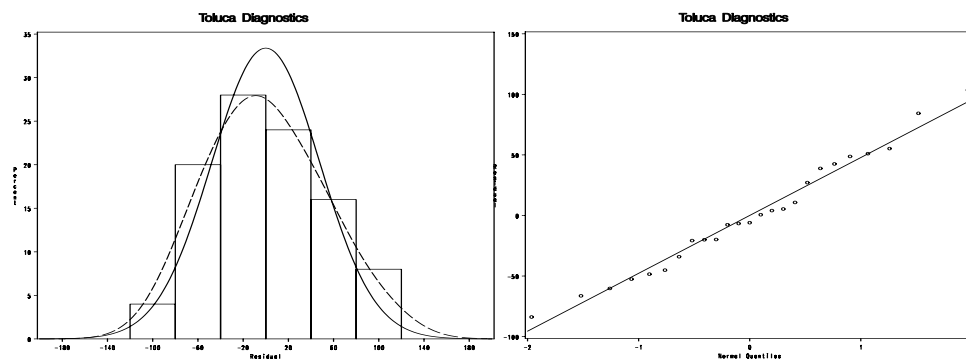
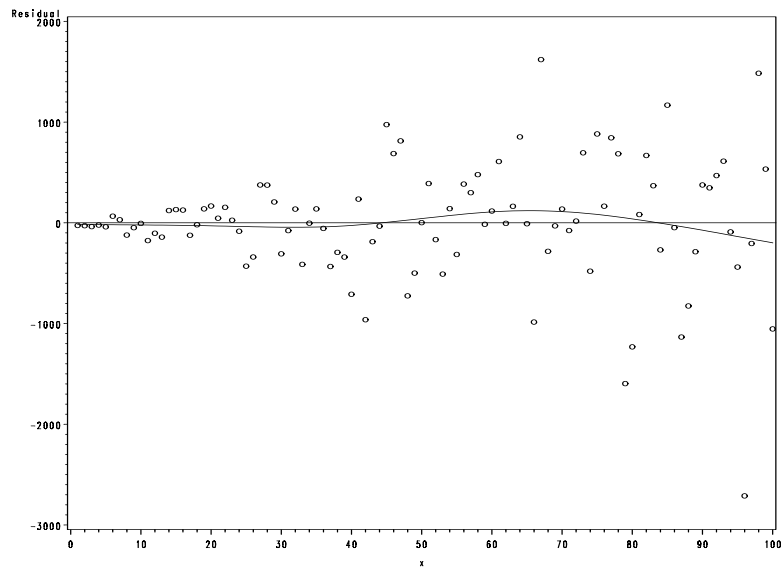
- H_0 : data are an iid sample from a normal population
- H_a : data are not an iid sample from a normal population
- KNNL (page 115) suggest a correlation test that requires a table look-up
- We have several choices for a significance testing procedure.
- `proc univariate` with the `normal` option provides four tests.

```
proc univariate data = diag normal;
    var resid;
```

Shapiro-Wilk is a common choice.

For Toluca:

Tests for Normality			
Test	--Statistic--		-----p Value-----
Shapiro-Wilk	W	0.978904	Pr < W 0.8626



Kolmogorov-Smirnov	D	0.09572	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.033263	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq	0.207142	Pr > A-Sq	>0.2500

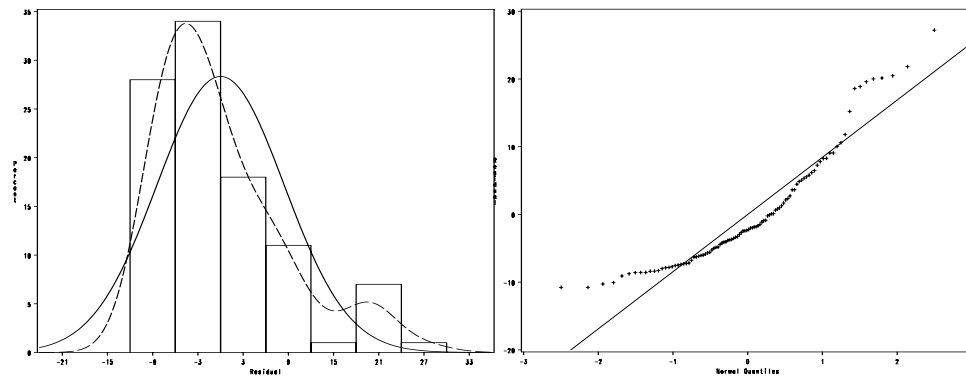
For expo:

Tests for Normality

Test	--Statistic--	-----p Value-----
Shapiro-Wilk	W 0.885197	Pr < W <0.0001
Kolmogorov-Smirnov	D 0.14061	Pr > D <0.0100
Cramer-von Mises	W-Sq 0.550032	Pr > W-Sq <0.0050
Anderson-Darling	A-Sq 3.470824	Pr > A-Sq <0.0050

Are There Outliers?

- Plot Y vs X
- Plot e vs X
- Plot of e vs X should emphasize an outlier



Simulate data from the model

$$Y = 30 + 50X + N(0, 200^2),$$

but with one observation made really large at $X = 50$. (knn1105b.sas)

```
data outlier50;
  do x=1 to 100 by 5;
    y=30+50*x+200*normal(0);
    output;
  end;
  x=50; y=30+50*50 +10000;
  d='out'; output;
```

Analyze without the outlier

```
proc reg data=outlier50;
  model y=x;
  where d ne 'out';
```

Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	-48.91638	81.76914	-0.60	0.5571
x	1	49.66468	1.44923	34.27	<.0001
Root MSE		186.86063	R-Square	0.9849	

Analyze with the outlier

```
proc reg data=outlier50;
  model y=x;
  output out=a2 r=resid;
```

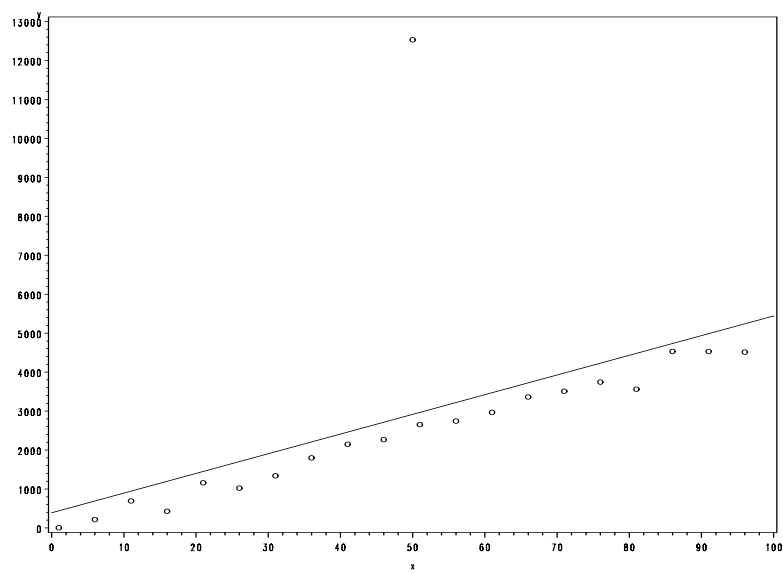
Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	-48.91638	81.76914	-0.60	0.5571
x	1	49.66468	1.44923	34.27	<.0001
Root MSE		186.86063	R-Square	0.9849	

Intercept	1	389.69949	987.07462	0.39	0.6974
x	1	50.53208	17.58446	2.87	0.0097
Root MSE		2267.45084	R-Square	0.3030	

```

symbol1 v=circle i=r1;
proc gplot data=a2;
  plot y*x/frame;

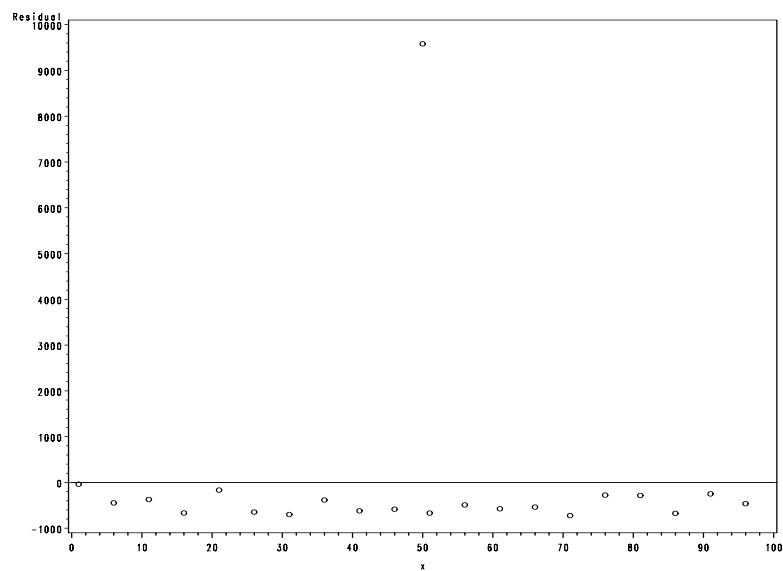
```



```

proc gplot data=a2;
  plot resid*x / vref=0;

```



Different Kinds of Outliers

The outlier in the last example *influenced* the intercept but not the slope. It inflated all of our standard errors.

Here is an example of an outlier that influences the slope. Use the model

$$Y = 30 + 50X + N(0, 200^2)$$

with one data point made really small at $X = 100$. (`knn1105c.sas`)

```
data outlier100;
  do x=1 to 100 by 5;
    y=30+50*x+200*normal(0);
    output;
  end;
  x=100; y=30+50*100 -10000;
  d='out'; output;
```

Analysis without the outlier

```
proc reg data=outlier100;
  model y=x;
  where d ne 'out';
```

Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	151.52255	110.59369	1.37	0.1875
x	1	47.45220	1.96010	24.21	<.0001
Root MSE		252.73113	R-Square	0.9702	

Analysis with the outlier included (save the residuals)

```
proc reg data=outlier100;
  model y=x;
  output out=a2 r=resid;
```

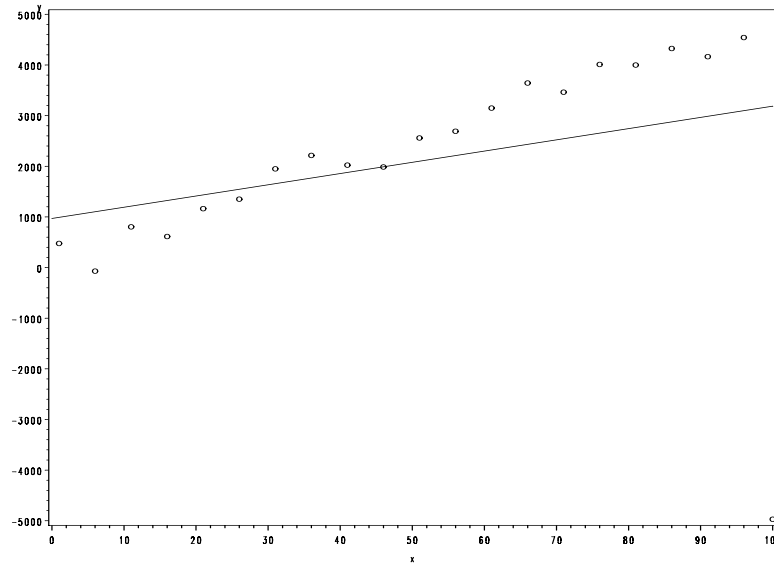
Parameter Estimates					
Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > t
Intercept	1	969.23278	887.17305	1.09	0.2883
x	1	22.18243	14.97865	1.48	0.1550
Root MSE		2072.84983	R-Square	0.1035	

```
symbol1 v=circle i=r1;
```

```
proc gplot data=a2;
  plot y*x;
```

```
proc gplot data = a2;
  plot resid*x/ vref = 0;
```

(See next page)



Dependent Errors

We may see this in a plot of residuals vs time order (KNNL) or sequence in the file.

We may see, for example, trends and/or cyclical effects.

Other, more subtle dependences may be more difficult to detect, especially since the information needed to detect it may not be included with the dataset. Ideally, the problem of dependence is handled at the experimental design stage, so that any dependence is either eliminated or explicitly included in the data.

See KNNL pages 108-109.

Summary of Diagnostics

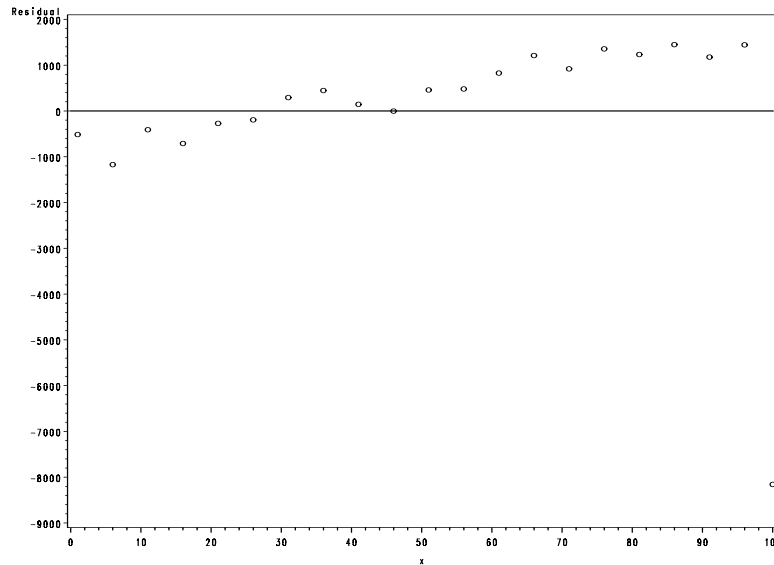
You will have noticed that the same plots are used for checking more than one assumption. These are your basic tools. **The following plots should be examined for every regression you ever do for the rest of your life, whether or not they are specifically asked for.**

- Plot Y vs X (check for linearity, outliers)
- Plot residuals vs X (check for constant variance, outliers, linearity, normality, patterns)
- qqplot and/or histogram of residuals (normality)

If possible, consider also doing a sequence plot of the residuals, but if the data are given in X order, this will not be informative.

Plots vs significance tests

If you are uncertain what to conclude after examining the plots, you may additionally wish to perform hypothesis tests for model assumptions (normality, homogeneity of variance,



independence). These tests are not a replacement for plots, but rather a *supplement* to them.

- Plots are more likely to suggest a remedy.
- Significance tests results are very dependent on the sample size; with sufficiently large samples, we can reject most null hypotheses.
- Significance tests are designed to possibly reject H_0 : non-rejection does not mean that H_0 is necessarily true (although if the test has good power, H_0 may be “true enough”)

Other Tests for Model Assumptions

- Durbin-Watson test for serially correlated errors (KNNL, page 114)
- Brown-Forsythe test for homogeneity of variance (KNNL, pages 116-117)
- Breusch-Pagan test for homogeneity of variance (KNNL, page 118)
- For SAS commands for residual tests, see `knnl114.sas`

We have discussed how to examine plots to detect departures from the important assumptions.

- linear relationship
- constant variance
- normal errors
- (independence)...

Now let's see what we can do about it.

Remedial Measures

Nonlinear relationships

We can model many nonlinear relationships with linear models; some have several explanatory variables (we'll see how to do this with multiple linear regression). Some examples include

$$\begin{aligned} Y &= \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon \text{ (quadratic)} \\ Y &= \beta_0 + \beta_1 \log(X) + \epsilon \end{aligned}$$

In these examples, we simply consider X , X^2 , and $\log(X)$ as individual quantities. That is to say that we consider the model $Y = \beta_0 + \beta_1 X'$, where $X' = \log(X)$. The model is still linear in terms of the regression coefficients. Other times it is necessary to transform a nonlinear equation into a linear equation. Consider the log-linear example:

$$Y = \beta_0 e^{\beta_1 X + \epsilon}$$

We can form a linear model taking logs:

$$\log(Y) = \log(\beta_0) + \beta_1 X + \epsilon$$

or

$$Y' = \beta'_0 + \beta_1 X + \epsilon$$

Note that assumptions about the error term also change with transformations. We could also perform a nonlinear regression analysis; beyond the scope of this course.

- KNNL, Chapter 13
- SAS `proc nlin`

Alternatively – guess at a good transformation and see if it works.

Nonconstant Variance

Sometimes we can model the way in which error variance changes

- may be linearly related to X

We can then use a weighted analysis, aka weighted linear regression.

- Use a `weight` statement in `proc reg`
- This is covered KNNL 11.1, so we will leave this until later.

Nonnormal Errors

Transformations often help; these are called *variance-stabilizing transformations*.

More complicated solution if transformations don't work: if we know what distribution the error terms have, model that explicitly, using a procedure that allows different distributions for the error term: SAS `proc genmod`.

`genmod`

We will not focus on this, but here are some distributions of Y that `genmod` can handle:

- Binomial (Yes/No or two-category data)
- Poisson (Count data)
- Gamma (exponential)
- Inverse gaussian
- Negative binomial
- Multinomial
- Specify a link function of $E(Y)$.

For now we will focus on transformations.

Transformations

Basic framework:

1. If the residuals appear to be normal with constant variance, and the relationship is linear, then go ahead with the regression model.
2. If the residuals appear to be normal with constant variance, but the relationship is non-linear, try transforming the X 's to make it a straight line.
3. If the residuals are badly-behaved, try transforming Y . If that stabilizes the variance but wrecks the straight line, try transforming X as well to get the straight line back.
4. Transformations might simultaneously fix problems with residuals and linearity (and normality).

Remember that if you choose a transformation, you need to go back and do all the diagnostics all over again.

If a Y transformation doesn't fix a non-constant variance, weighted least squares might work.

What Transformation to Use?

Pictures of various prototype situations are given in Section 3.9.

Figure 3.13, page 130: transformations on X for non-linear relationships

Figure 3.15, page 132: transformations on Y for non-constant variance (and possibly non-linear relationships)

Choosing a good transformation is a skill that improves with experience. A good way to get that experience is to try a lot of transformations on a lot of different datasets, and look at the graphs to see what happens.

One semi-automated way of choosing a transformation for Y is to use the Box-Cox procedure, which suggests a transformation on Y . (Yes, you still need to check all the diagnostics all over again.)

Box-Cox Procedure

Transformations on Y can sometimes be used to adjust for nonnormality and nonconstant variance.

If we restrict to a certain class called “power transformations”, there is an automated procedure that can help figure out which transformations might help.

Box-Cox will give us a “suggestion” of what type of transformation to try. Eventually, you would probably suggest the same transformation “by eye”, but this is especially good before you gain experience; also it may suggest a transformation you didn’t think of.

Box-Cox examines transformations that are either powers of Y or the natural log of Y . If we denote the new (transformed) Y as Y' , then the Box-Cox procedure will result in $Y' = Y^\lambda$ or $Y' = \ln(Y)$.

Important Special Cases

$\lambda = 1$, $Y' = Y^1$, no transformation

$\lambda = 0.5$, $Y' = Y^{1/2} = \sqrt{Y}$, square root

$\lambda = -0.5$, $Y' = Y^{-1/2} = \frac{1}{\sqrt{Y}}$, reciprocal square root

$\lambda = -1$, $Y' = Y^{-1}$, reciprocal

$\lambda = 0$, $Y' = \ln(Y)$ = (natural) log of Y (doesn’t follow the Y^λ formula; we just define it that way)

Box-Cox Details

We can estimate λ by including it as a parameter in a nonlinear model

$$Y^\lambda = \beta_0 + \beta_1 X + \epsilon$$

Details are in KNNL, pages 134-135.

Detailed “by-hand” Box-Cox transformation SAS code is in `knn1134.sas`

Automated Box-Cox procedure in `proc transreg`, illustrated in `boxcox.sas`.

It *suggests* a transformation, but there is no guarantee it will solve all your problems; still have to check residuals, assumptions, etc.

Helpful Details for Understanding knnl134.sas

Standard transformed Y is

$K_1(Y^\lambda - 1)$ if $\lambda \neq 0$

$K_2 \log(Y)$ if $\lambda = 0$,

where $K_2 = (\prod Y_i)^{1/n}$ (the geometric mean)
and $K_1 = \frac{1}{\lambda K_2^{\lambda-1}}$.

Run regression lots of times with different values of λ . Try to minimize SSE; maximize R^2 and likelihood.

Have a look at knnl134.sas to see how it works.

An Example

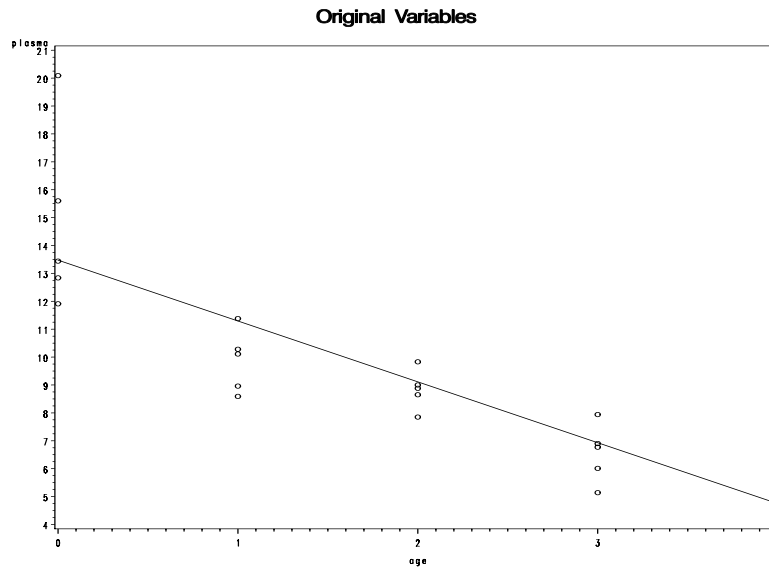
Let's use the automated procedure in boxcox.sas

```
data orig; input age plasma @@;
cards;
  0 13.44 0 12.84 0 11.91 0 20.09 0 15.60
  1 10.11 1 11.38 1 10.28 1 8.96 1 8.59
  2 9.83 2 9.00 2 8.65 2 7.85 2 8.88
  3 7.94 3 6.01 3 5.14 3 6.90 3 6.77
  4 4.86 4 5.10 4 5.67 4 5.75 4 6.23
;
* First let's look at the scatterplot to see the relationship;
title1 'Original Variables';
proc print data=orig;
symbol1 v=circle i=rl;
proc gplot data=orig;
  plot plasma*age;

proc reg data=orig;
  model plasma=age;
  output out = notrans r = resid;

Root MSE          1.84135      R-Square      0.7532

symbol1 i=sm70;
proc gplot data = notrans;
  plot resid*age / vref = 0;
```



```
proc univariate data=notrans;
  var resid; qqplot/normal (L=1 mu = est sigma=est);
```

* The residuals do not appear to have constant variance and the
 * relationship is not quite linear. Use the Box-Cox procedure to
 * suggest a possible transformation of the Y variable;

```
proc transreg data = orig;
  model boxcox(plasma)=identity(age);
```

```

The TRANSREG Procedure
Transformation Information
for BoxCox(plasma)

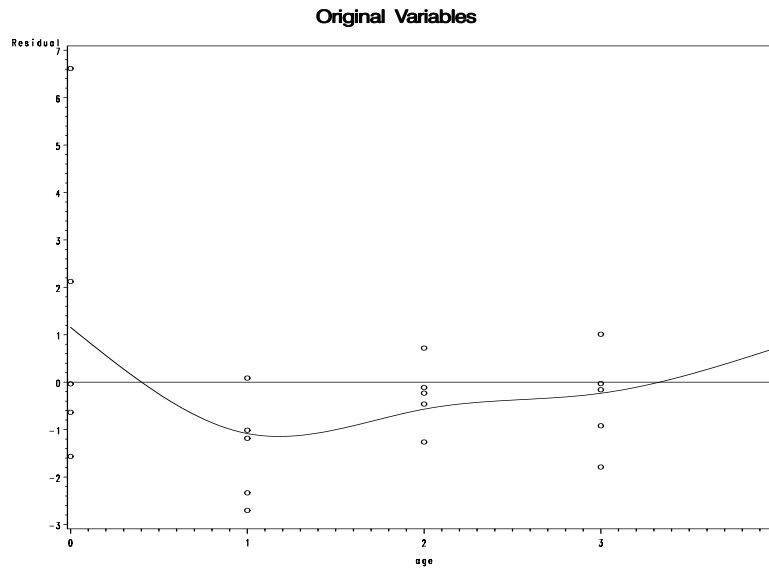
```

Lambda	R-Square	Log Like
-2.00	0.80	-12.3665
-1.75	0.82	-10.1608
-1.50	0.83	-8.1127
-1.25	0.85	-6.3056
-1.00	0.86	-4.8523 *
-0.75	0.86	-3.8891 *
-0.50	0.87	-3.5523 <
-0.25	0.86	-3.9399 *
0.00 +	0.85	-5.0754 *
0.25	0.84	-6.8988
0.50	0.82	-9.2925
0.75	0.79	-12.1209
1.00	0.75	-15.2625

```

< - Best Lambda
* - Confidence Interval
+ - Convenient Lambda

```



* Box-Cox suggests $\log Y$ or $1/\sqrt{Y}$. Let's do both of these.;

```

title1 'Transformed Variables';
data trans; set orig;
    logplasma = log(plasma);
    rsplasma = plasma**(-0.5);
proc print data = trans;
run;
title1 'Log Transformation';
proc reg data = trans;
    model logplasma = age;
    output out = logtrans r = logresid;

```

Root MSE	0.14385	R-Square	0.8535
----------	---------	----------	--------

```

symbol1 i=rl;
proc gplot data = logtrans;
    plot logplasma * age;

```

```

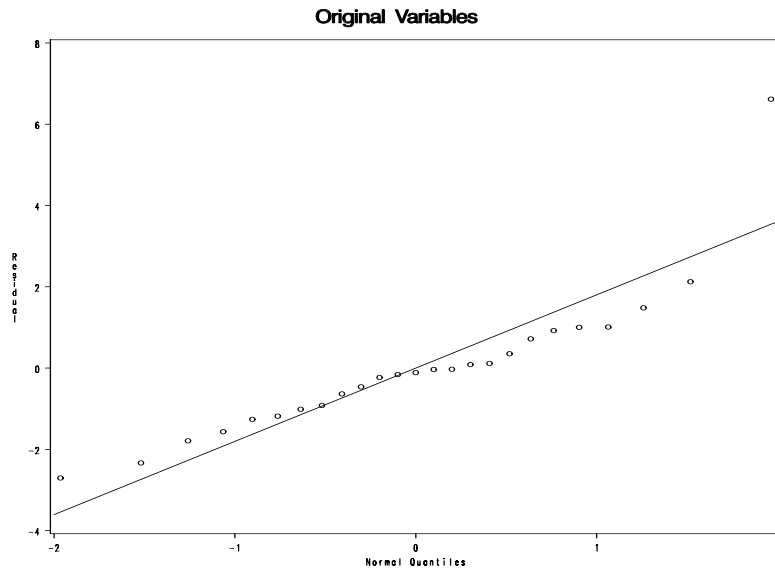
symbol1 i=sm70;
proc gplot data = logtrans;
    plot logresid * age / vref = 0;

```

```

proc univariate data=logtrans;
    var logresid;
    qqplot/normal (L=1 mu = est sigma = est);

```



```

title1 'Reciprocal Square Root Transformation';
proc reg data = trans;
    model rsplasma = age;
    output out = rstrans r = rsresid;

```

```

Root MSE          0.02319    R-Square    0.8665

```

```

symbol1 i=rl;
proc gplot data = rstrans;
    plot rsplasma * age;

```

```

symbol1 i=sm70;
proc gplot data = rstrans;
    plot rsresid * age / vref = 0;

```

```

proc univariate data=rstrans;
    var rsresid; qqplot/normal (L=1 mu = est sigma = est);

```

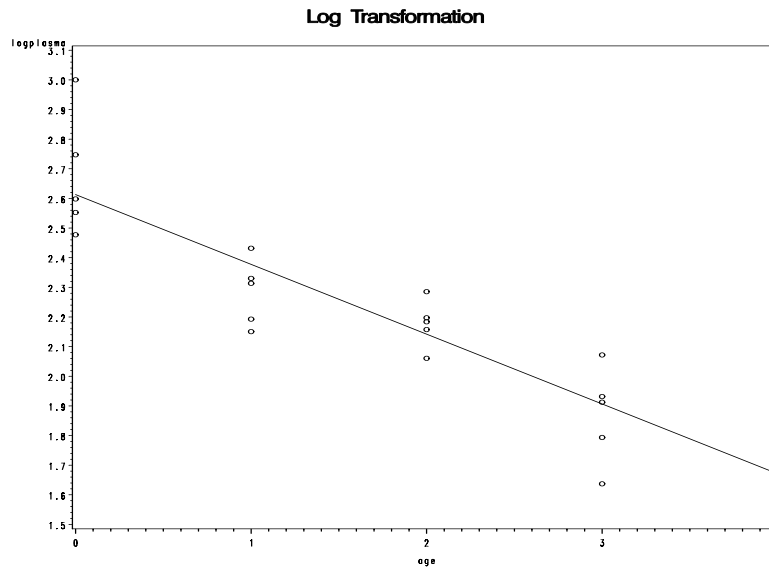
We went over KNNL 3.8 and 3.9.

Sections 3.4-3.7 – Description of significance tests for assumptions (read it if you are interested).

Summary of Remedial Measures

Nonlinear Relationships

Sometimes a transformation on X will fix this. It could involve, for example, both linear and quadratic terms in X , for which we need multiple regression.



Nonconstant Variance

If we can model the way in which error variance changes, we can use weighted regression. We'll talk about this later, when we get to KNNL 11.1.

Use a **weight** statement in **proc reg**.

Sometimes a transformation on Y will work instead.

Nonnormal Errors

Transformations on Y often help.

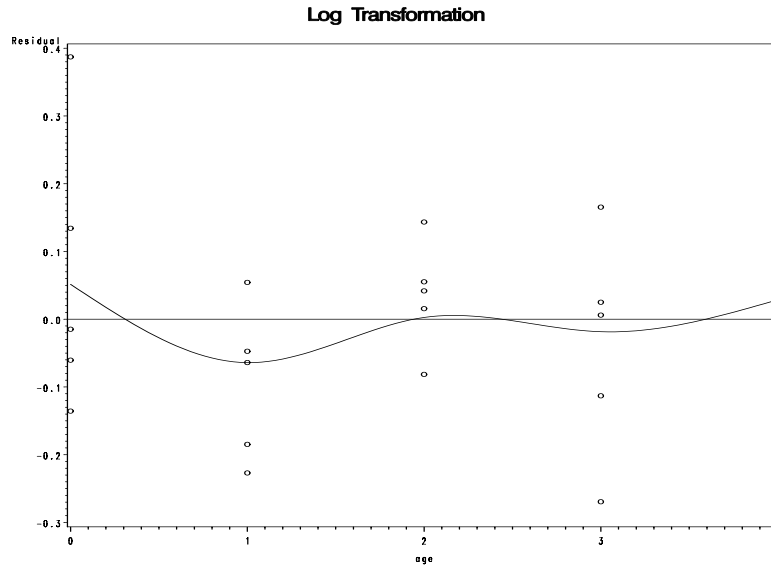
Could use a procedure that allows different distributions for the error term: SAS **proc genmod** (not covered here).

Box-Cox Transformations

Suggests some possible Y transformations to try. Still have to try them out and look at the graphs to see if they really do fix your problems. Sometimes the “best” transformation is still not good enough. Also, sometimes a transformation only improves things a tiny bit, in which case it may be better to go with the untransformed variable, because it is much easier to interpret.

Other Topics (KNNL, Chapter 4)

- Joint estimation of β_0 and β_1
- Multiplicity
- Regression through the origin



- Measurement error
- Inverse predictions

Joint Estimation of β_0 and β_1

Confidence intervals are used for a single parameter.

Confidence regions (e.g., confidence band) for two or more parameters

The region for (β_0, β_1) defines a set of lines.

Since b_0 and b_1 are (jointly) normal, the natural confidence region is an ellipse (STAT 524) (i.e., smallest region).

KNNL use rectangles (i.e. region formed from two intervals $a_1 \leq \beta_1 \leq a_2$ and $a_3 \leq \beta_0 \leq a_4$), using the Bonferroni Correction (KNNL 4.1).

Bonferroni Correction

We want the probability that both intervals are correct to be ≥ 0.95 .

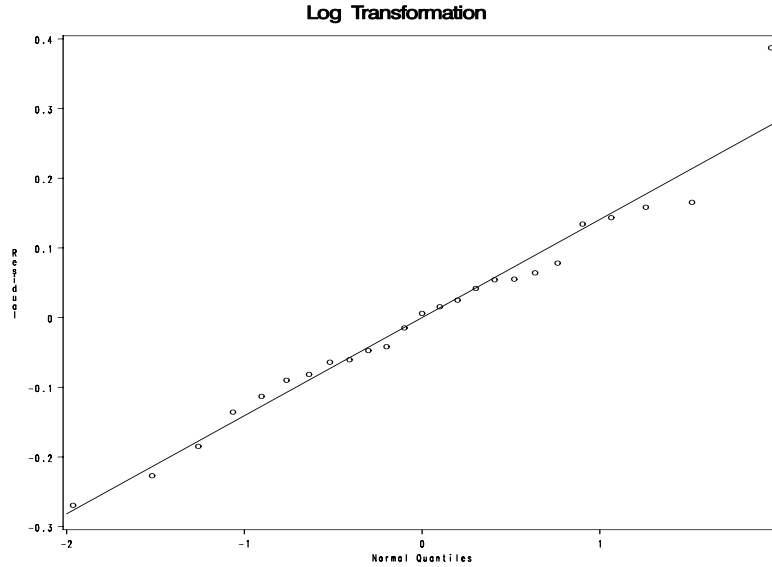
- Basic idea is that we have an error budget ($\alpha = 0.05$), so spend half on β_0 and half on β_1
- We use $\alpha = 0.025$ for each CI (97.5% CI), leading to

$$b_0 \pm t_c s\{b_0\}$$

$$b_1 \pm t_c s\{b_1\}$$

where $t_c = t_{n-2}(0.9875)$. Note $0.9875 = 1 - 0.025/2$.

- We start with a 5% error budget, and we have two intervals so we give 2.5% to each.
- Each interval has two ends (tails) so we again divide by 2.



Theory Behind this Correction: Bonferroni Inequality

Let the two intervals be I_1 and I_2 .

We will call it “COR” if the interval contains the correct parameter value, “INC” if not.

$$\begin{aligned}
 P(\text{both COR}) &= 1 - P(\text{at least one INC}) \\
 P(\text{at least one INC}) &= P(I_1 \text{ INC}) + P(I_2 \text{ INC}) - P(\text{both INC}) \\
 &\leq P(I_1 \text{ INC}) + P(I_2 \text{ INC}). \\
 \text{Thus, } P(\text{both COR}) &\geq 1 - P(I_1 \text{ INC}) - P(I_2 \text{ INC})
 \end{aligned}$$

If we use $0.05/2$ for each interval, $1 - (P(I_1 \text{ INC}) + P(I_2 \text{ INC})) = 1 - 0.05 = 0.95$.

So $P(\text{both correct})$ is *at least* 0.95 (it is better if they happen to overlap).

We will use this same idea when we do multiple comparisons in ANOVA.

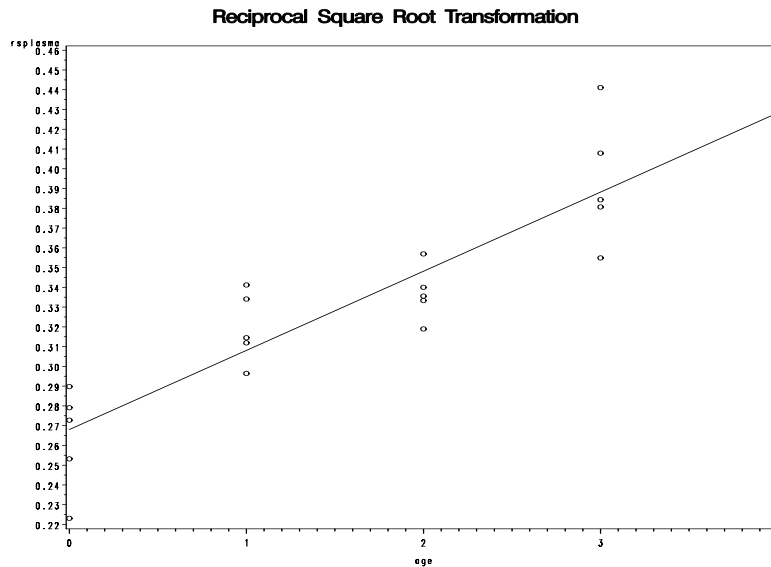
Mean Response CI's

We already talked about simultaneous estimation for all X_h with a confidence band: use Working-Hotelling (KNNL 2.6).

$$\hat{Y}_h \pm Ws\{\hat{Y}_h\}, \text{ where } W^2 = 2F_{2,n-2}(1 - \alpha)$$

For simultaneous estimation for a few X_h , say g different values, we may use Bonferroni instead.

$$\hat{Y}_h \pm Bs\{\hat{Y}_h\}, \text{ where } B = t_{n-2}(1 - \alpha/(2g))$$



Simultaneous PI's

Simultaneous prediction intervals for g different X_h : use Bonferroni.

$$\hat{Y}_h \pm Bs\{pred\}, \text{ where } B = t_{n-2}(1 - \alpha/(2g))$$

or Scheffé

$$\hat{Y}_h \pm Ss\{pred\}, \text{ where } S^2 = gF_{g,n-2}(1 - \alpha).$$

Regression Through the Origin

$$Y_i = \beta_1 X_i + \epsilon_i$$

`noint` option in `proc reg`.

Generally *not* a good idea

Problems with r^2 and other statistics

See cautions, KNNL page 164

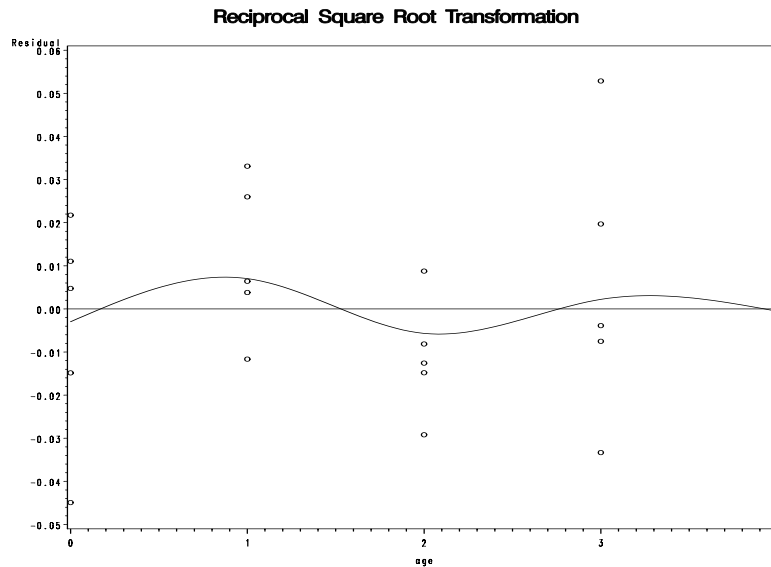
Measurement Error

For Y , this is usually not a problem (taken care of by ϵ)

For X , we can get biased estimators of our regression parameters

See KNNL 4.5, pages 165-167.

Berkson model: special case where measurement error in X is no problem.



Inverse Prediction

Sometimes called *calibration*.

Given Y_h , predict the corresponding value of X ,

Solve the fitted equation for X_h

$$\hat{X}_h = \frac{(Y_h - b_0)}{b_1}, b_1 \neq 0$$

Approximate CI can be obtained, see KNNL, page 168.

We did Chapter 4 fairly quickly (important part – Bonferroni inequality)

Next we will do simple regression with vectors and matrices so that we can generalize to multiple regression. Look at KNNL 5.1 to 5.7. If you have ever had a course in linear algebra, this should be familiar to you. If not, you will need to read these sections in detail. We will not cover them in class, but I will be happy to answer questions in office hours.

