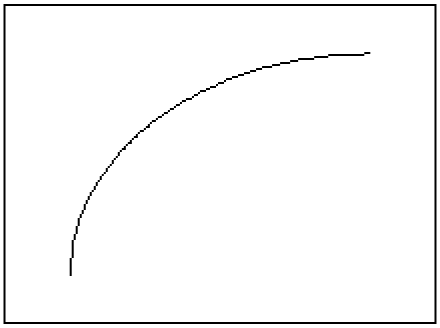
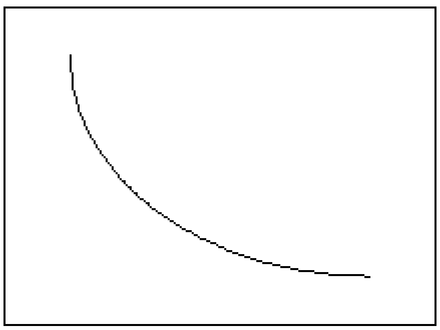
Shape of Scatterplot and Possible Choices of Transformations

1



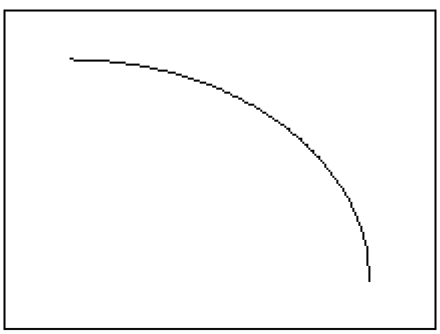
x	or	y
log x		y^2
$\frac{1}{x}$		y^3
	etc.	

2



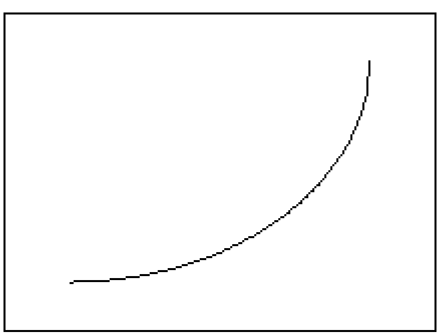
x	or	y
log x		log y
$\frac{1}{x}$		$\frac{1}{y}$
	etc.	

3



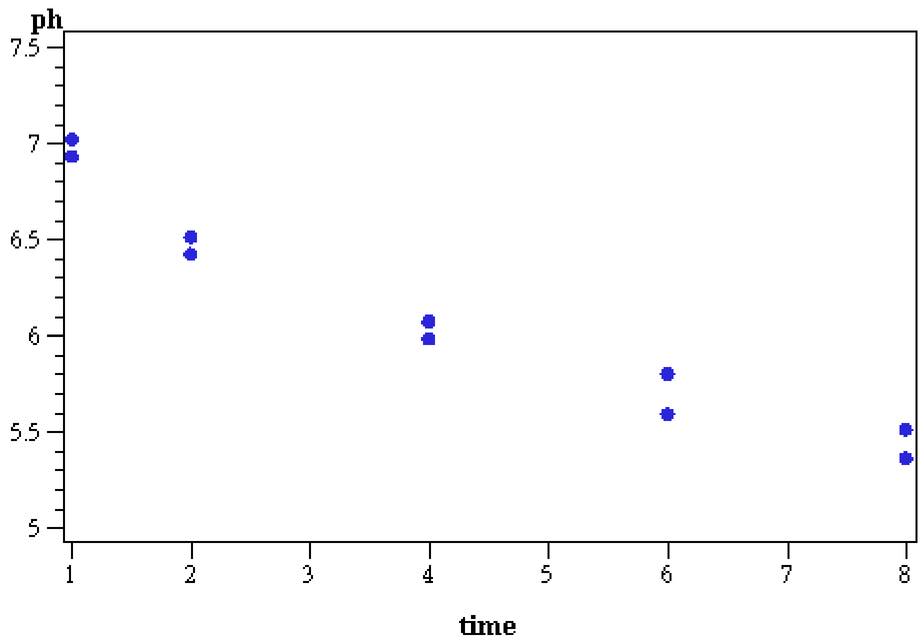
x	or	y
x^2		y^2
x^3		y^3
	etc	

4



x	or	y
x^2		log y
x^3		$\frac{1}{y}$
	etc.	

```
options ls=78 ps=45 nodate nonumber;
data phstudy;
 input time ph @@;  * '@@' tells SAS to keep reading data on this line;
 datalines;
 1 7.02 1 6.93 2 6.42 2 6.51 4 6.07 4 5.98 6 5.59 6 5.80 8 5.51 8 5.36
 ;

 proc gplot data=phstudy;
  plot ph*time;
  run;
```



The shape of the scatterplot looks like graph #2 on the previous page.

We could try to get a more accurate model by transforming time (x variable) using either a logarithmic or reciprocal transformation, or we could transform pH (y variable) using either of these two transformations.

First, we ignore the curvature and fit a simple linear model.

```
title1 'pH regressed on time';
 proc reg data=phstudy;
  model ph=time /p r;
  output out=newph predicted=pred residual=resid;
  run;
```

pH regressed on time

The REG Procedure
Model: MODEL1
Dependent Variable: ph

Number of Observations Read	10
Number of Observations Used	10

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	2.85612	2.85612	110.29	<.0001
Error	8	0.20717	0.02590		
Corrected Total	9	3.06329			

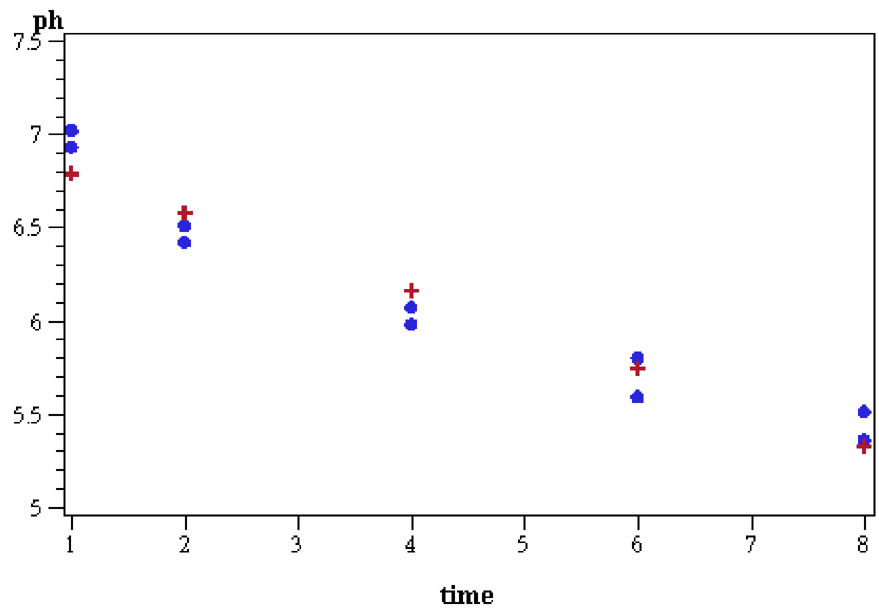
Root MSE	0.16092	R-Square	0.9324
Dependent Mean	6.11900	Adj R-Sq	0.9239
Coeff Var	2.62990		

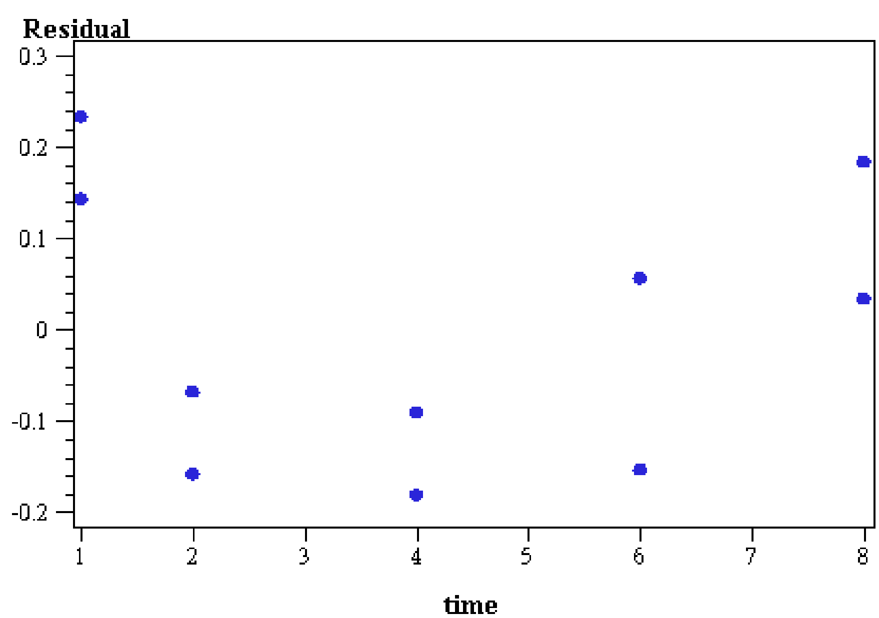
Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	6.99537	0.09774	71.57	<.0001
time	1	-0.20866	0.01987	-10.50	<.0001

Plot the observed points and the fitted values from simple linear regression. We know that this is probably a bad fit.

```
proc gplot data=newph;
  plot ph * time
       pred * time / overlay;
  plot resid * time;
  run;
```





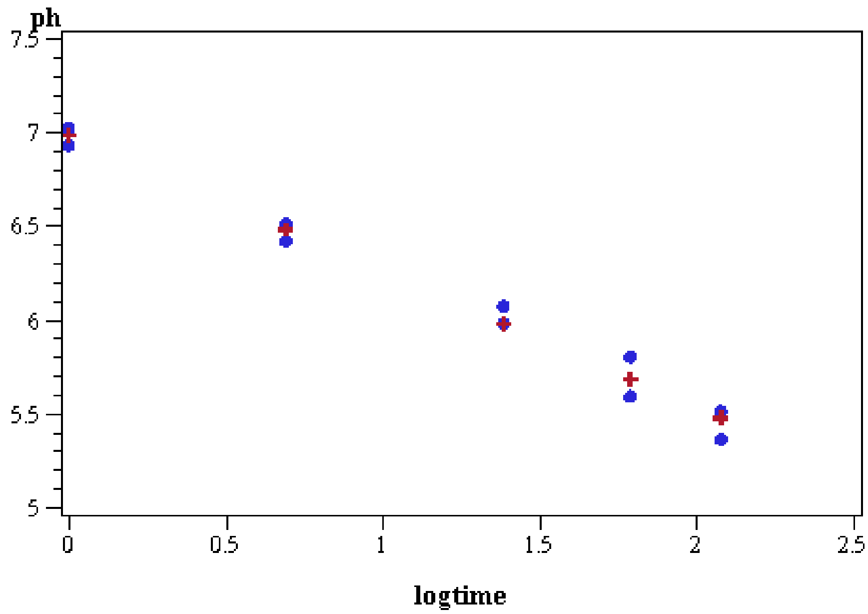
We will follow one of the recommended transformations and try to get a better fit.

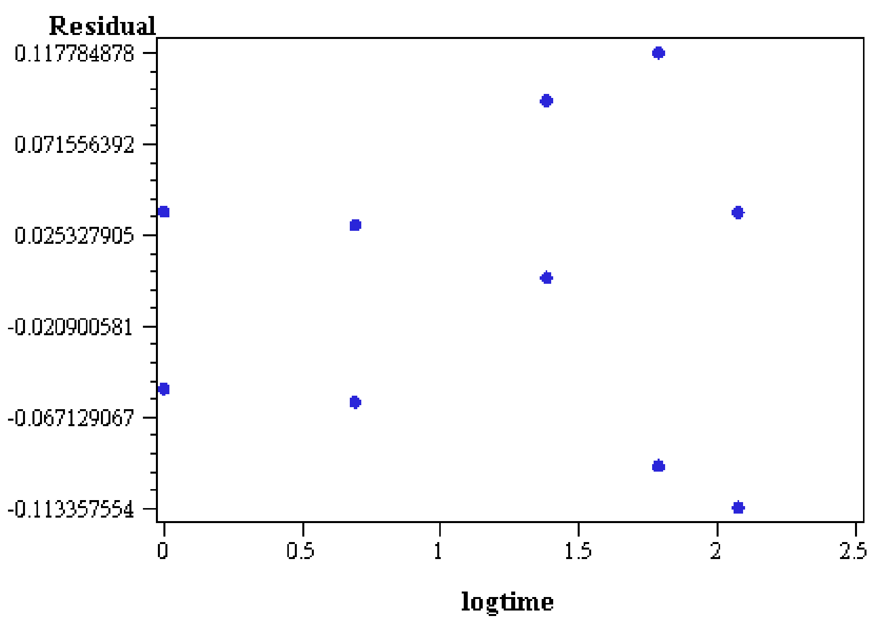
Transform Time (X) and Generate a New Model

```
data transform;
 set phstudy;  * this reads in all the data from the 'phstudy' data set;
 logtime = log(time);  * log transform X and save it as a new variable;
 run;

 title1 'pH regressed on log(time)';
 proc reg data=transform;
  model ph=logtime /p r;
  output out=newph2 predicted=pred residual=resid;
  run;
```

```
                              pH regressed on log(time)

                                  The REG Procedure
                                    Model: MODEL1
                               Dependent Variable: ph

                       Number of Observations Read          10
                       Number of Observations Used          10


                                Analysis of Variance

                                      Sum of           Mean
Source                   DF          Squares         Square    F Value    Pr > F

Model                     1          3.00931        3.00931     446.03    <.0001
Error                     8          0.05398        0.00675
Corrected Total           9          3.06329


             Root MSE              0.08214    R-Square     0.9824
             Dependent Mean        6.11900    Adj R-Sq     0.9802
             Coeff Var             1.34237


                                Parameter Estimates

                             Parameter       Standard
     Variable      DF         Estimate          Error    t Value    Pr > |t|

     Intercept      1          6.98303        0.04846     144.10      <.0001
     logtime        1         -0.72600        0.03438     -21.12      <.0001
```

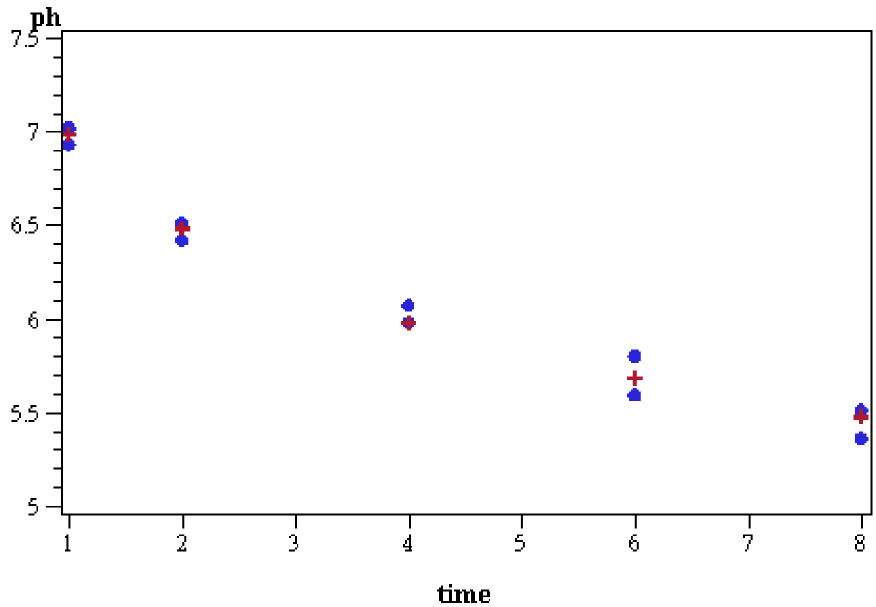
```
proc gplot data=newph2;
  plot ph * logtime
       pred * logtime / overlay;
  plot resid * logtime;
  run;
```



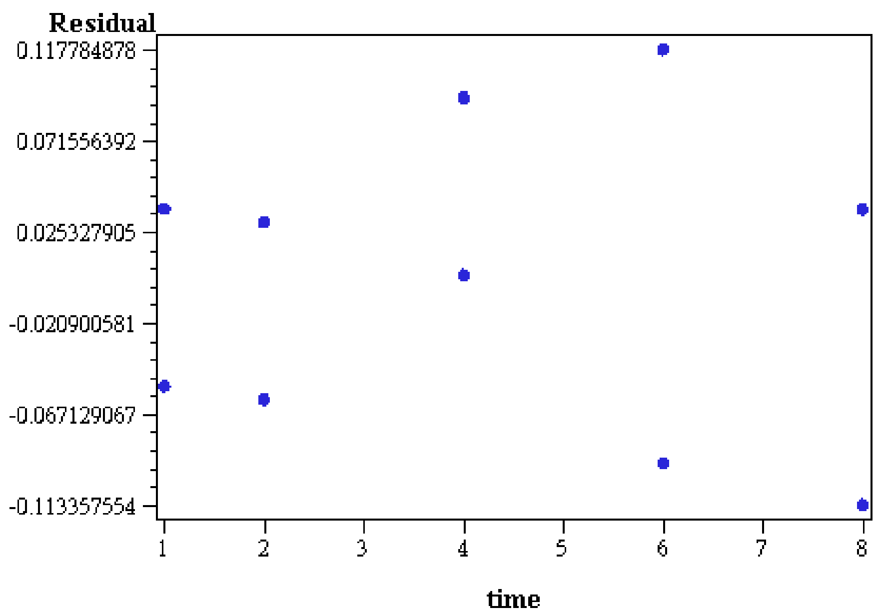


But we don't want our supervisor to deal with "log time". We need to graph the results using "time".

```
/* graph the results using time instead of logtime */
title 'Back-Transformed "Time"';
proc gplot data=newph2;
  plot ph *time
       pred * time / overlay;
  plot resid *time;
  run;
```



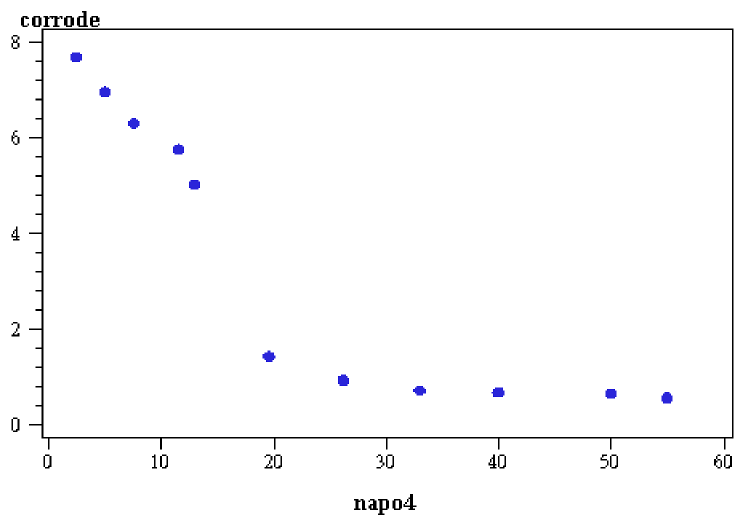
Notice how the fitted "linear" model bends to fit the points.



Compare these plots to the original simple linear model.

A New Example: Metal corrosion as a function of the concentration of a chemical.

```
options ls=78 ps=60 nodate nonumber;
data corrosion;
 input napo4 corrode @@;
 datalines;
  2.5 7.68      5.03 6.95      7.6 6.3      11.6 5.75     13 5.01     19.6 1.43
 26.2 0.93     33.0  0.72     40    0.68     50   0.65     55 0.56
 ;

 title 'Corrosion Data';
 proc gplot data=corrosion;
 plot corrode * napo4;
 run;
```



Some combination of these transformations might be useful: log x, log y, $\frac{1}{x}$ and/or $\frac{1}{y}$

There are 9 possible models with these variables. We will keep track of them in this table.

Enter R^2 and RMSE for each model	y	log y	$\frac{1}{y}$
x			
log x			
$\frac{1}{x}$			

Fit a Bunch of Models Using Original and Transformed Variables

```
* create a new data set that contains the transformed variables;
 data transform;
  set corrosion;
  log_napo4 = log(napo4);
  log_corrode = log(corrode);
  inv_napo4 = 1/napo4;
  inv_corrode = 1/corrode;
  run;

* run a bunch of models;
  proc reg data=transform;
  model corrode = napo4;          * Model 1: original variables;
  output out=new predicted=pred residual=resid;
  model log_corrode = napo4;      * Model 2: log Y;
  model inv_corrode = napo4;      * Model 3: 1/Y;
  model corrode = inv_napo4;      * Model 4: 1/X;
  model corrode = log_napo4;      * Model 5: log X;
  model log_corrode = log_napo4; * Model 6: log Y and log X;
  model inv_corrode = inv_napo4; * Model 7: 1/Y and 1/X;
  run;
```

The REG Procedure
Model: MODEL1
Dependent Variable: corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	67.84940	67.84940	30.94	0.0004
Error	9	19.73701	2.19300		
Corrected Total	10	87.58642			

Root MSE	1.48088	R-Square	0.7747
Dependent Mean	3.33273	Adj R-Sq	0.7496
Coeff Var	44.43444		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	6.73510	0.75731	8.89	<.0001
napo4	1	-0.14202	0.02553	-5.56	0.0004

The REG Procedure
Model: MODEL2
Dependent Variable: log_corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	10.72150	10.72150	56.62	<.0001
Error	9	1.70428	0.18936		
Corrected Total	10	12.42578			

Root MSE	0.43516	R-Square	0.8628
Dependent Mean	0.70353	Adj R-Sq	0.8476
Coeff Var	61.85337		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	2.05603	0.22254	9.24	<.0001
napo4	1	-0.05645	0.00750	-7.52	<.0001

The REG Procedure
Model: MODEL3
Dependent Variable: inv_corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	4.17887	4.17887	142.91	<.0001
Error	9	0.26317	0.02924		
Corrected Total	10	4.44203			

Root MSE	0.17100	R-Square	0.9408
Dependent Mean	0.79678	Adj R-Sq	0.9342
Coeff Var	21.46130		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-0.04760	0.08745	-0.54	0.5994
napo4	1	0.03525	0.00295	11.95	<.0001

The REG Procedure
Model: MODEL4
Dependent Variable: corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	57.52465	57.52465	17.22	0.0025
Error	9	30.06177	3.34020		
Corrected Total	10	87.58642			

Root MSE	1.82762	R-Square	0.6568
Dependent Mean	3.33273	Adj R-Sq	0.6186
Coeff Var	54.83859		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	1.28437	0.73979	1.74	0.1166
inv_napo4	1	20.93677	5.04509	4.15	0.0025

The REG Procedure
Model: MODEL5
Dependent Variable: corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	78.66723	78.66723	79.38	<.0001
Error	9	8.91919	0.99102		
Corrected Total	10	87.58642			

Root MSE	0.99550	R-Square	0.8982
Dependent Mean	3.33273	Adj R-Sq	0.8869
Coeff Var	29.87045		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	11.24080	0.93697	12.00	<.0001
log_napo4	1	-2.81318	0.31575	-8.91	<.0001

The REG Procedure
Model: MODEL6
Dependent Variable: log_corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	10.78507	10.78507	59.16	<.0001
Error	9	1.64071	0.18230		
Corrected Total	10	12.42578			

Root MSE	0.42697	R-Square	0.8680
Dependent Mean	0.70353	Adj R-Sq	0.8533
Coeff Var	60.68888		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	3.63163	0.40187	9.04	<.0001
log_napo4	1	-1.04163	0.13542	-7.69	<.0001

The REG Procedure
Model: MODEL7
Dependent Variable: inv_corrode

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	1.95687	1.95687	7.09	0.0260
Error	9	2.48517	0.27613		
Corrected Total	10	4.44203			

Root MSE	0.52548	R-Square	0.4405
Dependent Mean	0.79678	Adj R-Sq	0.3784
Coeff Var	65.95065		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	1.17458	0.21270	5.52	0.0004
inv_napo4	1	-3.86157	1.45057	-2.66	0.0260

We want small values for RMSE and large values for R^2. Based on these two measures of model fit, we should take a closer look at models 3 and 5.

Model 3 uses the reciprocal of Y with the original X. RMSE = 0.17 R^2 = 0.94

Model 5 uses the original Y with the log of X. RMSE = 1.0 R^2 = 0.90

WE CANNOT COMPARE THESE MODELS USING <u>RMSE</u> BECAUSE THEY HAVE DIFFERENT Y'S.

Transforming the Y generates new challenges in interpreting the results of the fitted model, so we take a closer look at Model 3.

```
data orig_scale;
 set model3;
 corrode_hat = 1/inv_pred;       * reciprocal of predicted values;
 upper_hat = 1/uclm;             * reciprocal of upper confidence limit;
 lower_hat = 1/lclm;             * reciprocal of lower confidence limit;
 diff = corrode - corrode_hat;   * 'true' residuals;
 diffsq = diff**2;

proc print data=orig_scale;
 var napo4 corrode inv_corrode inv_pred corrode_hat
     lower_hat upper_hat diff diffsq;
 run;

proc gplot data=orig_scale;
 plot corrode_hat * napo4
      corrode * napo4 /overlay;
 run;
```

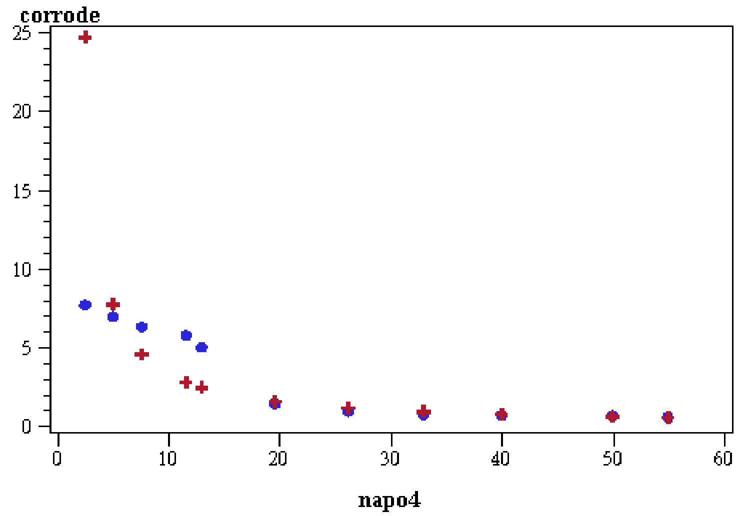
Corrosion Data

Obs	napo4	corrode	inv_ corrode	inv_pred	corrode_ hat	upper_ hat	lower_ hat	diff	diffsq
1	2.50	7.68	0.13021	0.04051	24.6846	4.44195	-6.9394	-17.0046	289.157
2	5.03	6.95	0.14388	0.12968	7.7112	3.31622	-23.7051	-0.7612	0.579
3	7.60	6.30	0.15873	0.22026	4.5401	2.63183	16.5124	1.7599	3.097
4	11.60	5.75	0.17391	0.36124	2.7682	1.98391	4.5781	2.9818	8.891
5	13.00	5.01	0.19960	0.41059	2.4355	1.82408	3.6637	2.5745	6.628
6	19.60	1.43	0.69930	0.64320	1.5547	1.30992	1.9120	-0.1247	0.016
7	26.20	0.93	1.07527	0.87582	1.1418	1.00663	1.3189	-0.2118	0.045
8	33.00	0.72	1.38889	1.11549	0.8965	0.80206	1.0161	-0.1765	0.031
9	40.00	0.68	1.47059	1.36221	0.7341	0.65769	0.8306	-0.0541	0.003
10	50.00	0.65	1.53846	1.71466	0.5832	0.51978	0.6643	0.0668	0.004
11	55.00	0.56	1.78571	1.89089	0.5289	0.46981	0.6049	0.0311	0.001

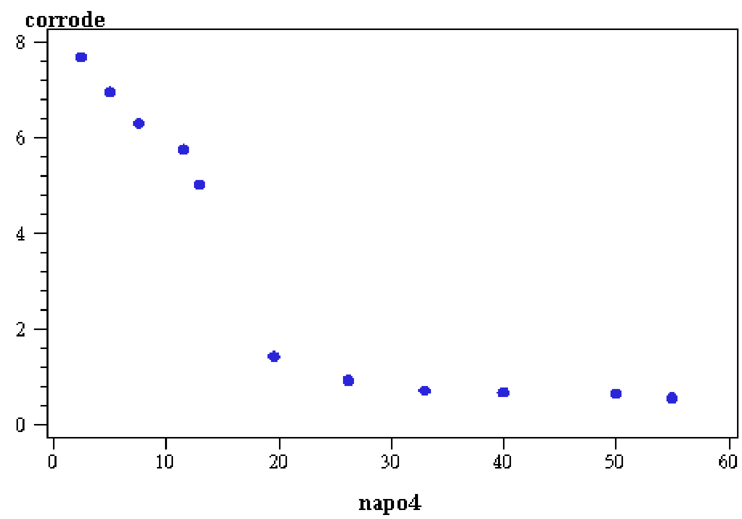
On the original scale,

solid blue circles: X = napo4 and Y = observed corrosion

red plus signs: X = napo4 and Y = predicted corrosion



For comparison, here is the original scatterplot.



The SAS code for this handout is on the course website.
The filename is TransformingXorY.sas.