

BIM 105

Probability and Statistics for Biomedical Engineers

David M. Rocke

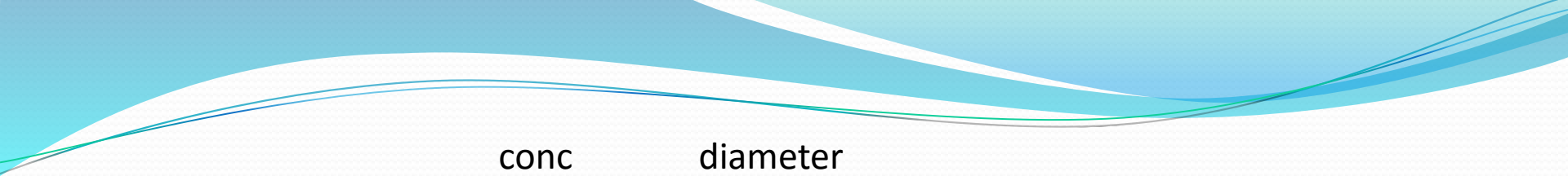
Department of Biomedical Engineering

Checking Assumptions

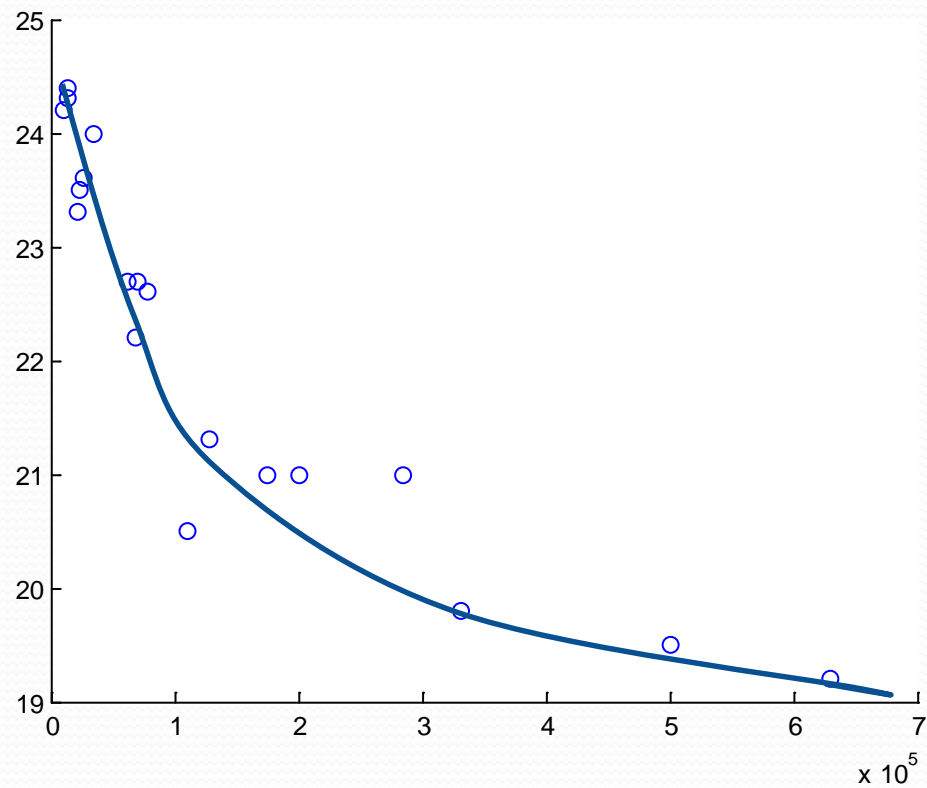
- The single best plot to check assumptions is residuals vs. fitted values.
- This can show nonlinearity through curvature.
- It can show heteroscedasticity through funnel-shaped residual plots.
- It is not as helpful for assessing independence.
- To some extent, it can find outliers and skewness.
- Consider taking logs of one or both of x and y .

Tetrahymena Experiment

- We examine an experiment on the growth of tetrahymena cells, a kind of protozoan sometimes used as a biological model.
- The cell concentration (conc) was set at the beginning of the experiment and the cells were allowed to grow.
- The average cell diameter (diameter) of the resulting cells was measured.
- Because of crowding, it was hypothesized that the greater the initial concentration, the smaller the diameter of the resulting cells.



conc	diameter
630000	19.2
501000	19.5
332000	19.8
285000	21.0
201000	21.0
175000	21.0
129000	21.3
111000	20.5
78000	22.6
70000	22.7
69000	22.2
62000	22.7
35000	24.0
27000	23.6
24000	23.5
22000	23.3
14000	24.4
13000	24.3
11000	24.2



Least Squares Fit

```
>> thlm = fitlm(conc,diameter)
```

Linear regression model:

$$y \sim 1 + x_1$$

Estimated Coefficients:

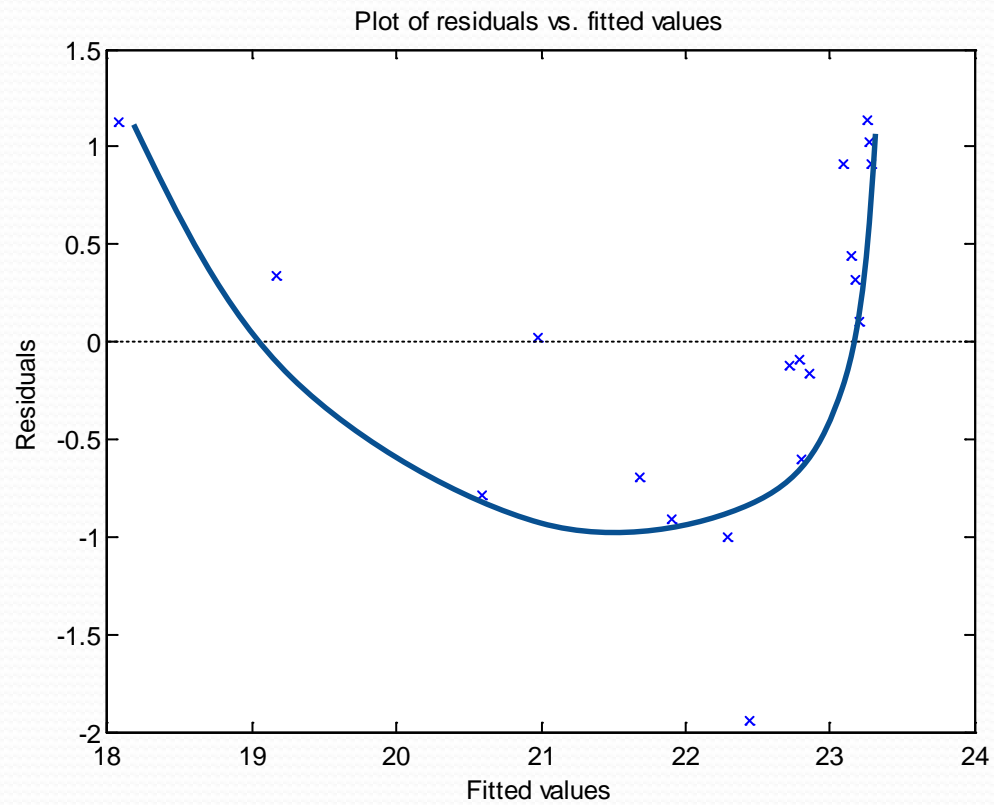
	Estimate	SE	tStat	pValue
(Intercept)	23.384	0.26269	89.015	3.8974e-24
x1	-8.4212e-06	1.1672e-06	-7.215	1.4457e-06

Number of observations: 19, Error degrees of freedom: 17

Root Mean Squared Error: 0.868

R-squared: 0.754, Adjusted R-Squared 0.739

F-statistic vs. constant model: 52.1, p-value = 1.45e-06



```
>> plotResiduals(thlm,'fitted')
>> lconc = log(conc)
>> ldiameter = log(diameter)
>> thllm = fitlm(lconc,ldiameter)
```

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7642	0.046533	80.893	1.9752e-23
x1	-0.059677	0.0041246	-14.468	5.4816e-11

Number of observations: 19, Error degrees of freedom: 17

Root Mean Squared Error: 0.0219

R-squared: 0.925, Adjusted R-Squared 0.92

F-statistic vs. constant model: 209, p-value = 5.48e-11

```
>> plotResiduals(thllm,'fitted')
>> scatter(lconc,ldiameter)
>> lsline
```


plotresiduals() options

'caseorder'	Residuals vs. case (row) order
'fitted'	Residuals vs. fitted values
'histogram'	Histogram
'lagged'	Residuals vs. lagged residual ($r(t)$ vs. $r(t-1)$)
'probability'	Normal probability plot
'symmetry'	Symmetry plot



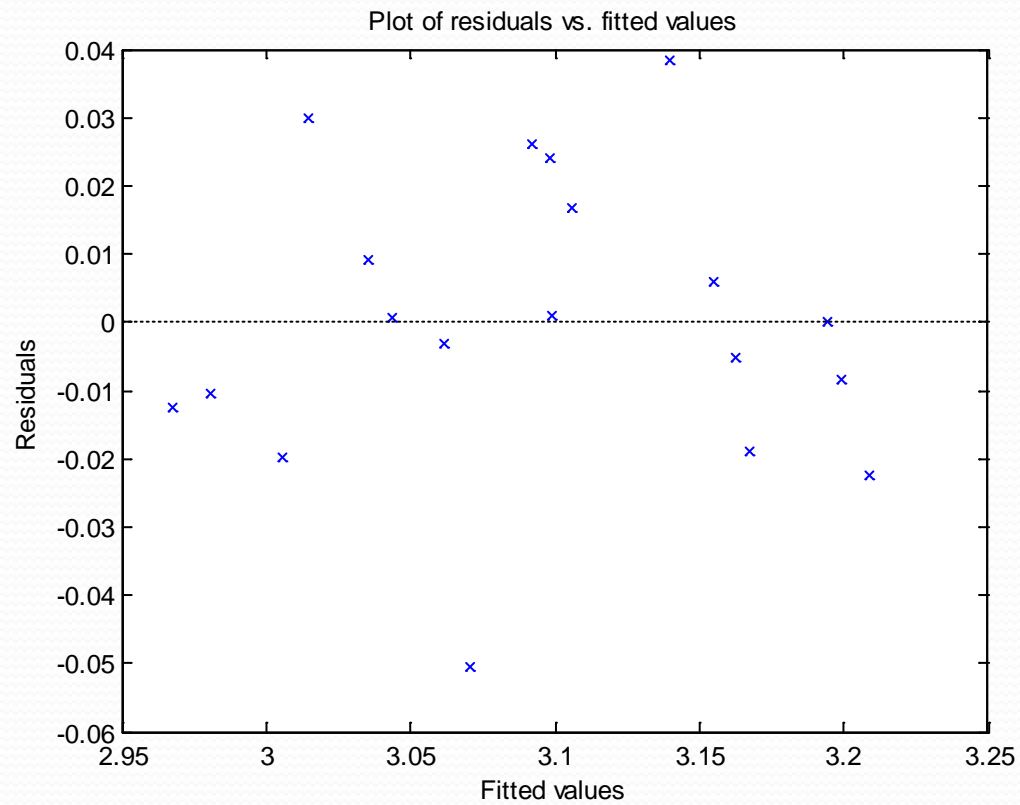
Most Useful

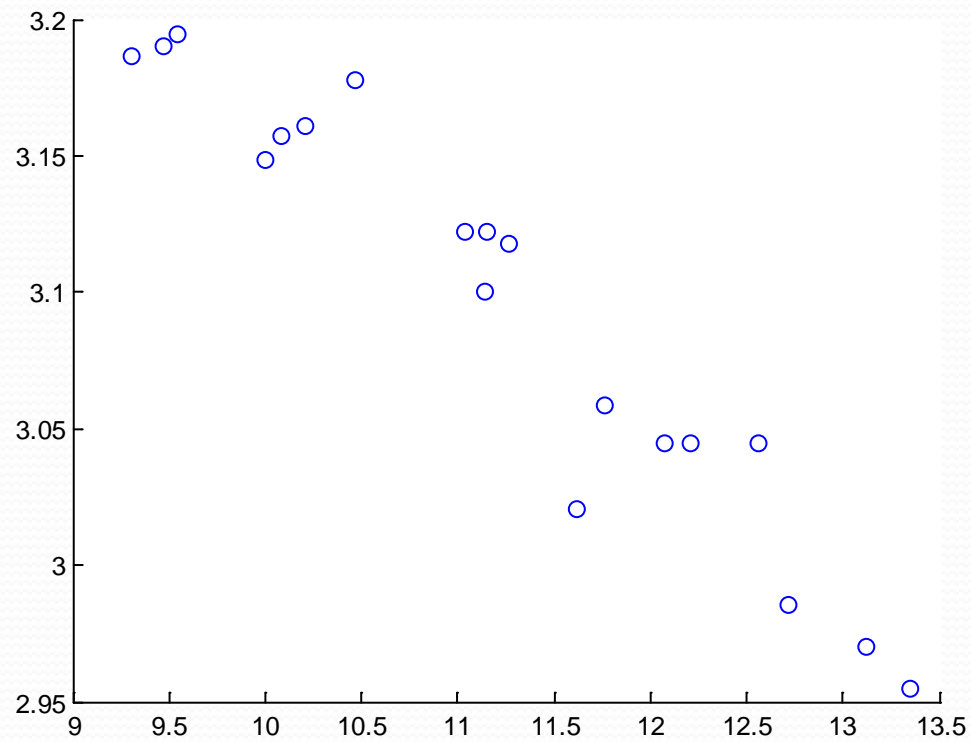


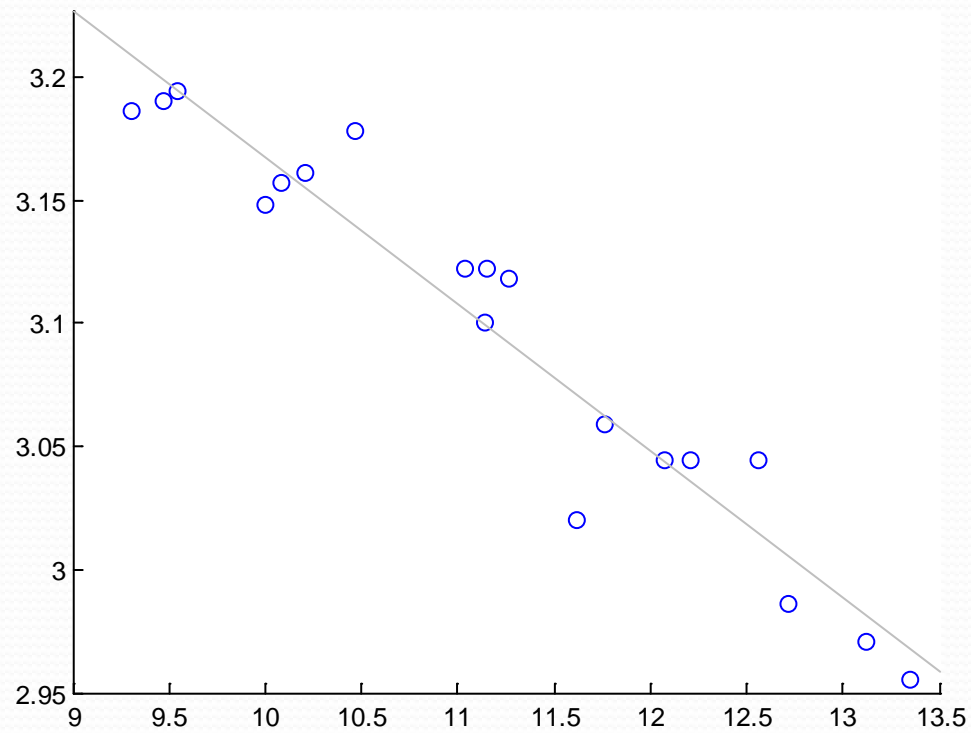
Default



Autocorrelation check







Tetrahymena Experiment

- The average cell diameter varies systematically with the starting concentration.
- The relationship is clearly nonlinear.
- Both variables are measurements, and the starting concentration ranges across orders of magnitudes (factor of 60).
- Taking the log of the diameter is not so much needed since the range is only 19-24.
- But a theoretical model suggests that $D=aC^b$ so $\text{Log}(D) = \log(a) + b \log(C)$
- Either model will be good empirically.

Data Transformations

- Often the log is good for measured data and the square root for count data (Poisson).
- If X is Poisson with parameter λ , then
 - $E(X) = \lambda$
 - $V(X) = \lambda$
- If $Y = \sqrt{X}$, then approximately
 - $E(Y) = 0.5\sqrt{\lambda}$
 - $V(Y) = 0.25$
- In a regression with different data points at different values of λ , this stabilizes the variance.

Other Transformations

- The square root is $f(x) = x^{0.5}$.
- We can also use other powers like $f(x) = x^\alpha$.
 - For $\alpha = 0.5$, this is the square root.
 - For $\alpha = -1$, this is the reciprocal.
- If we vary this a bit, so that $f(x) = (x^\alpha - 1)/\alpha$, then as α approaches 0, the function approaches $\ln(x)$, so the log is in this power transformation family.
- It is not important to find the 'optimal' transformation, only one that is 'good enough'.
- This can linearize the relationship and make the errors more nearly constant in variance.

Models Suggested by Physical Laws

- If we have the height and diameter of a series of trees, and want the volume of lumber generated, then we can approximate the volume as a cylinder, with $V = \pi(D/2)^2H$ or as a cone, with $V = (1/3)\pi(D/2)^2H$, so in either case, $\ln(V) = a + 2 \ln(D) + \ln(H)$, where $a = \ln(\pi/4)$ or $a = \ln(\pi/12)$
- This suggests a model with two predictors, and with volume, height, and diameter all on the log scale.
- It will only approximately be correct.

Length of a Spring under Load

- Hooke's Law states that the measured length of a spring under a force F is $L_o + kF$, where L_o is the length under o load and k is a stiffness constant.
- This law is only approximate, like many such laws, and applies only when the load is not too great. For example, the extended length of a spring cannot exceed the length of the wire that is coiled to make the spring.
- Nonetheless, this suggests a linear model for length, with the intercept as the length under no load, and the slope as the stiffness constant.

TABLE 8.1 Measured lengths of a spring under various loads

Weight (lb) x	Measured Length (in.) y	Weight (lb) x	Measured Length (in.) y
0.0	5.06	2.0	5.40
0.2	5.01	2.2	5.57
0.4	5.12	2.4	5.47
0.6	5.13	2.6	5.53
0.8	5.14	2.8	5.61
1.0	5.16	3.0	5.59
1.2	5.25	3.2	5.61
1.4	5.19	3.4	5.75
1.6	5.24	3.6	5.68
1.8	5.46	3.8	5.80

Linear Model for Hooke Data

```
>> hookelm = fitlm(Weight,Length)
```

Linear regression model:

$$y \sim 1 + x_1$$

Estimated Coefficients:

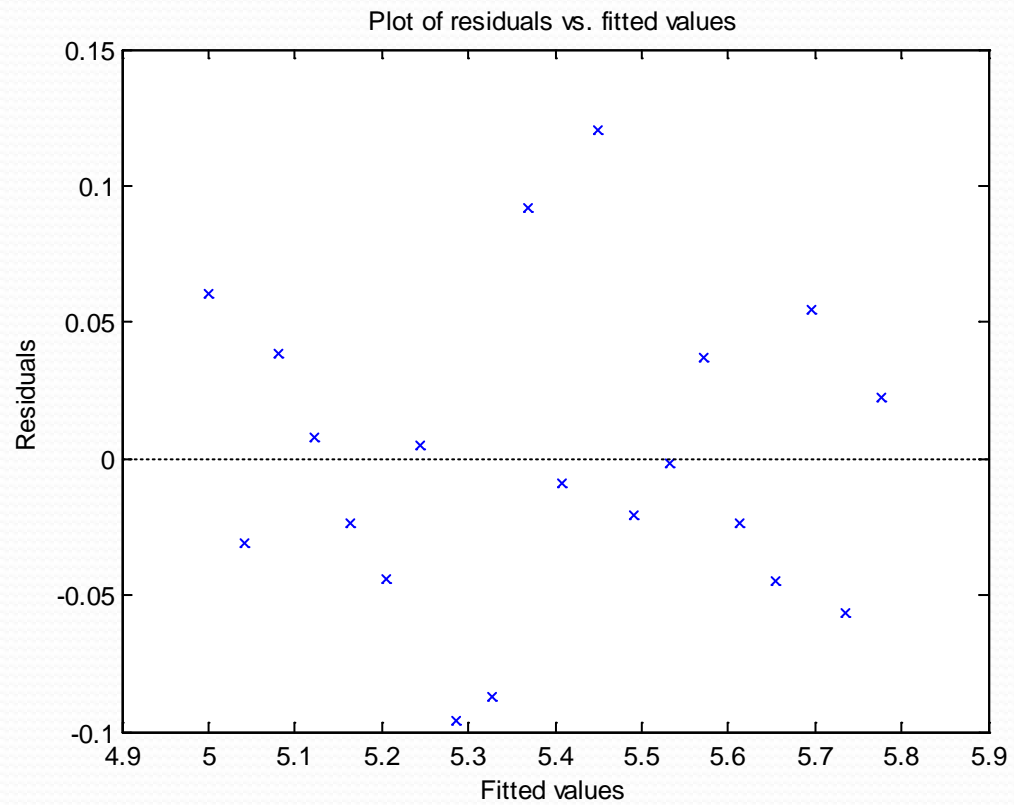
	Estimate	SE	tStat	pValue
(Intercept)	4.9997	0.024774	201.81	1.1884e-31
x1	0.20462	0.011146	18.358	4.2038e-13

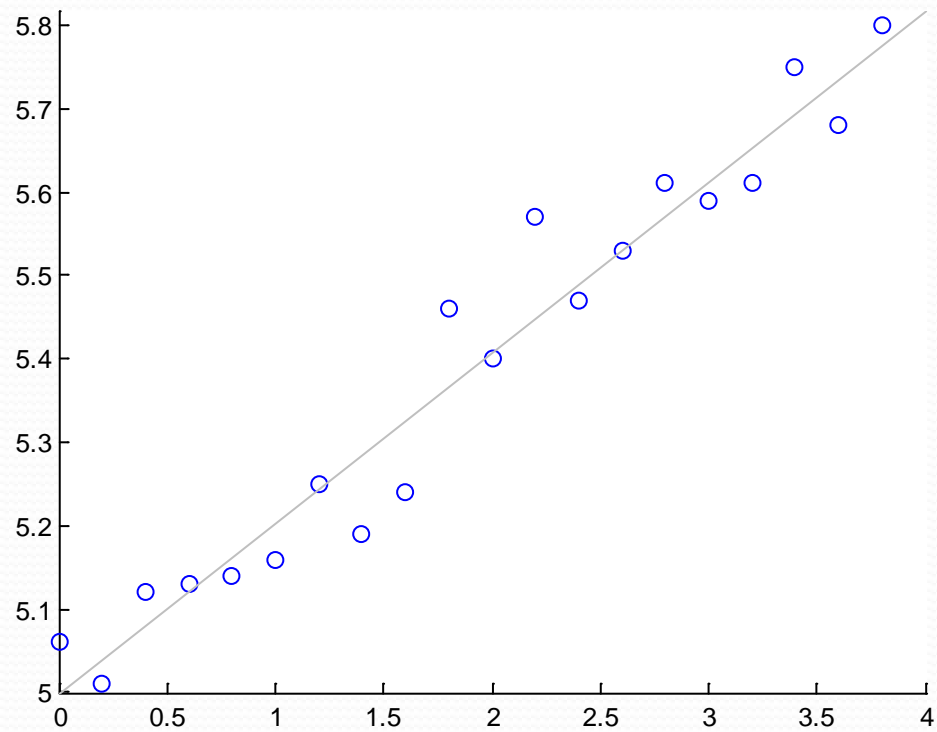
Number of observations: 20, Error degrees of freedom: 18

Root Mean Squared Error: 0.0575

R-squared: 0.949, Adjusted R-Squared 0.946

F-statistic vs. constant model: 337, p-value = 4.2e-13





Model Checking

```
>> plotResiduals(hookelm, 'fitted')
```

```
>> scatter(Weight, Length)
```

```
>> lsline
```

Multiple Regression

- We can use more than one predictor, in which case this is called multiple regression.
- The predictors can be quantitative or categorical.
- We can also use products (interactions) of variables, or even powers like x^2 .
- The computations are done by a computer almost always.
- The assumptions are essentially the same as for simple linear regression.

We would like to predict a quantitative variable y from predictors x_1, x_2, \dots, x_p

Each of these variables is either a quantitative variable (measurement or count) or a *dummy variable* with possible values 0 and 1.

We can have combinations like $x_2 = x_1^2$ (quadratic term) or $x_3 = x_1 x_2$ (interaction term).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p + \epsilon$$

Some special cases are the one-variable polynomial model

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_p x^p + \epsilon$$

and the two-variable quadratic model with interactions

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + \beta_4 x_1^2 + \beta_5 x_2^2 + \epsilon$$

These are still considered linear models because they are linear in the parameters

β_0, β_1, \dots even though they are nonlinear in the variables x_1, x_2, \dots unlike a model such as

$$y = \beta_0 + \beta_1 e^{\beta_2 x} + \epsilon$$

Any of these models can be fit by least squares giving coefficient estimates

Tetrahymena Experiment

- We examine an experiment on the growth of tetrahymena cells, a kind of protozoan sometimes used as a biological model.
- The cell concentration (conc) was set at the beginning of the experiment and the cells were allowed to grow.
- The average cell diameter (diameter) of the resulting cells was measured.
- Some runs had added glucose and some did not
- We want to examine the effects of glucose as well as starting concentration.

TH Data Regression

```
>> ldiameter = log(diameter)
>> lconc = log(conc)
>> th = table(lconc, glucose, ldiameter) #by default last column is response
>> thllm = fitlm(th)
```

Linear regression model:

```
ldiameter ~ 1 + lconc + glucose
```

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

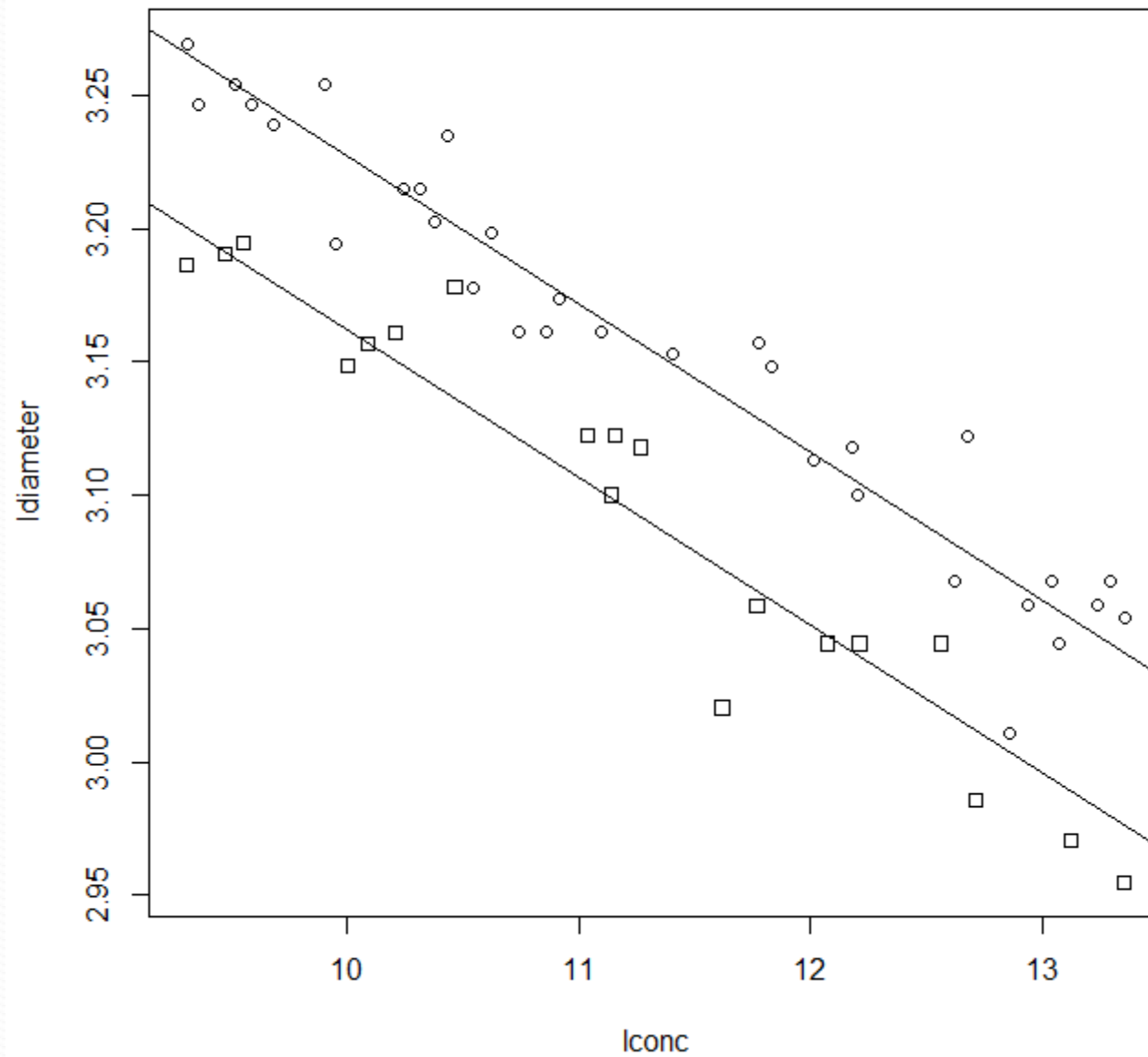
Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

lconc is log concentration. For every unit change in log concentration, there is a predicted change of -0.055393. A unit change in natural log concentration is a factor of 2.71 increase, with a predicted decrease to $\exp(-0.055393) = 0.946$ of the previous, or a 5.4% decrease.

The glucose variable is 1 if glucose is added and 0 otherwise, and the predicted increase is 0.06502 on the log scale or 6.7%.

Is it possible that the **rate of decrease in log diameter** with log concentration is different for experiments with glucose present than for experiments with glucose absent?



Model with Glucose (32 obs with glucose and 19 without)

Estimate	SE	tStat	pValue	
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Model without added glucose observations (just the original 19)

Estimate	SE	tStat	pValue	
(Intercept)	3.7642	0.046533	80.893	2.4631e-22
lconc	-0.059677	0.0041246	-14.468	1.3111e-10

```
>> thllm2 = fitlm(th,'ldiameter ~ lconc*glucose')
```

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7642	0.044221	85.123	3.8116e-53
lconc	-0.059677	0.0039197	-15.225	8.5488e-20
glucose	-0.0078692	0.054559	-0.14423	0.88593
lconc:glucose	0.0064805	0.0048209	1.3442	0.18532

The term for the interaction of glucose and log concentration is not significant, so the slopes are not significantly different. The slope estimate for no glucose is -0.059677 and for glucose it is $-0.059677 + 0.006480 = -0.053196$. The slope estimate without the interaction is -0.055393.

No Glucose	-0.060
Glucose	-0.053
Combined	-0.055

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7642	0.044221	85.123	3.8116e-53
lconc	-0.059677	0.0039197	-15.225	8.5488e-20
glucose	-0.0078692	0.054559	-0.14423	0.88593
lconc:glucose	0.0064805	0.0048209	1.3442	0.18532

The term for the interaction of glucose and log concentration is not significant, so the slopes are not significantly different. The interaction term is highly correlated with glucose (correlation = 0.98), and in this case, putting two correlated variables in the model makes both of them look non-significant.

The test for the coefficient is a test of whether eliminating that one variable would cause poorer predictions if all other variables are retained.

The model for prediction with interaction term is

$$\text{Log(diameter)} = \beta_0 + \beta_1 \log(\text{concentration}) + \beta_2 \text{glucose} + \beta_3 \log(\text{concentration}) * \text{glucose}$$

The variable glucose is 0 or 1.

When there is no glucose (19 observations) we predict

$$\text{Log(diameter)} = \beta_0 + \beta_1 \log(\text{concentration})$$

and the slope with respect to $\log(\text{concentration})$ is β_1

When there is glucose (32 observations) we predict

$$\text{Log(diameter)} = \beta_0 + \beta_1 \log(\text{concentration}) + \beta_2 + \beta_3 \log(\text{concentration})$$

and the slope with respect to $\log(\text{concentration})$ is $\beta_1 + \beta_3$

so a test of the hypothesis $H_0 : \beta_3 = 0$ is a test of whether the slope is the same with or without glucose

Without the interaction term, glucose contributes only to the intercept, not the slope


```
>> fitlm(th,'ldiameter ~ lconc + glucose')
```

	Estimate	SE	tStat	pValue
	<hr/>	<hr/>	<hr/>	<hr/>
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

The root mean square error $s = 0.021$ is an estimate of the sd around the regression line. $s^2 = 0.000442$.

$R^2 = 0.934$ is the $SS(\text{regression})/SS(\text{total})$. It is at most 1.

$F = 338$ is a statistical test of the hypothesis that none of the predictors is useful. It has an F distribution with 2 and 48 df.

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \epsilon$$

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$$

$$SST = \sum_{i=1}^n (y_i - \bar{y})^2$$

$$SSE = \sum_{i=1}^n (y_i - \hat{y})^2$$

$$SSR = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2$$

$$SST = SSR + SSE$$

$$s^2 = \frac{SSE}{n - p - 1}$$

$$R^2 = \frac{SST - SSE}{SST} = \frac{SSR}{SST}$$

$$F = \frac{SSR / p}{SSE / (n - p - 1)} = \frac{MSR}{MSE} \text{ this has an } F \text{ distribution with } p \text{ and } n - p - 1 \text{ df}$$

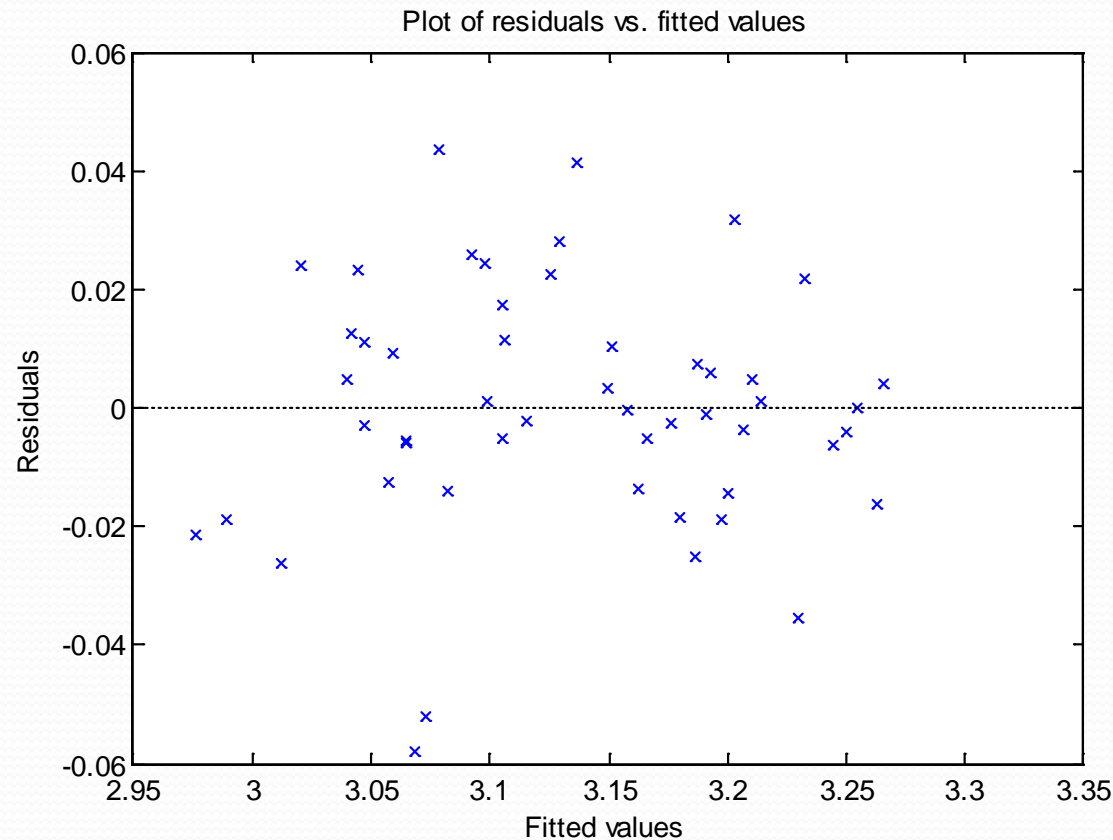
Predicted Values and Residuals

- As with univariate regression,
`[ldpred ldci] = predict(thllm)`
returns predicted values and a confidence interval.
- We can calculate the residuals by
`ldres = thllm.Residuals.Raw`
but often plotting them is sufficient.
- The main useful plot is of residuals vs. fitted values and this is produced by `plotResiduals(thllm, 'fitted')`.
- We can also plot residuals vs. each predictor as in
`scatter(lconc, ldres)`.
`boxplot(ldres, glucose)`

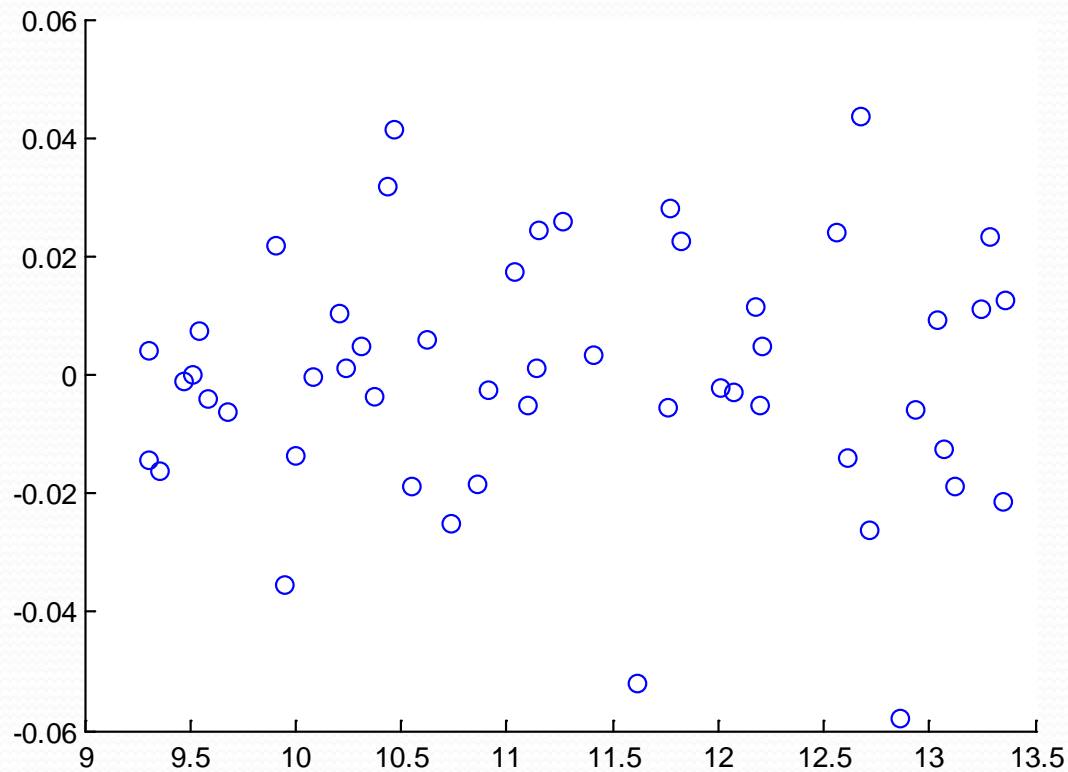
Model Checking

- Plot residuals vs. fitted values.
- Scatter plot of residuals vs. each quantitative predictor.
- Boxplot of residuals vs. each categorical predictor.
- If observations are time ordered, plot residuals against time, especially in cases where there is one observation per day/hour/month, etc.
- Look for curvature, changes in variance, outliers, and anything else unusual.

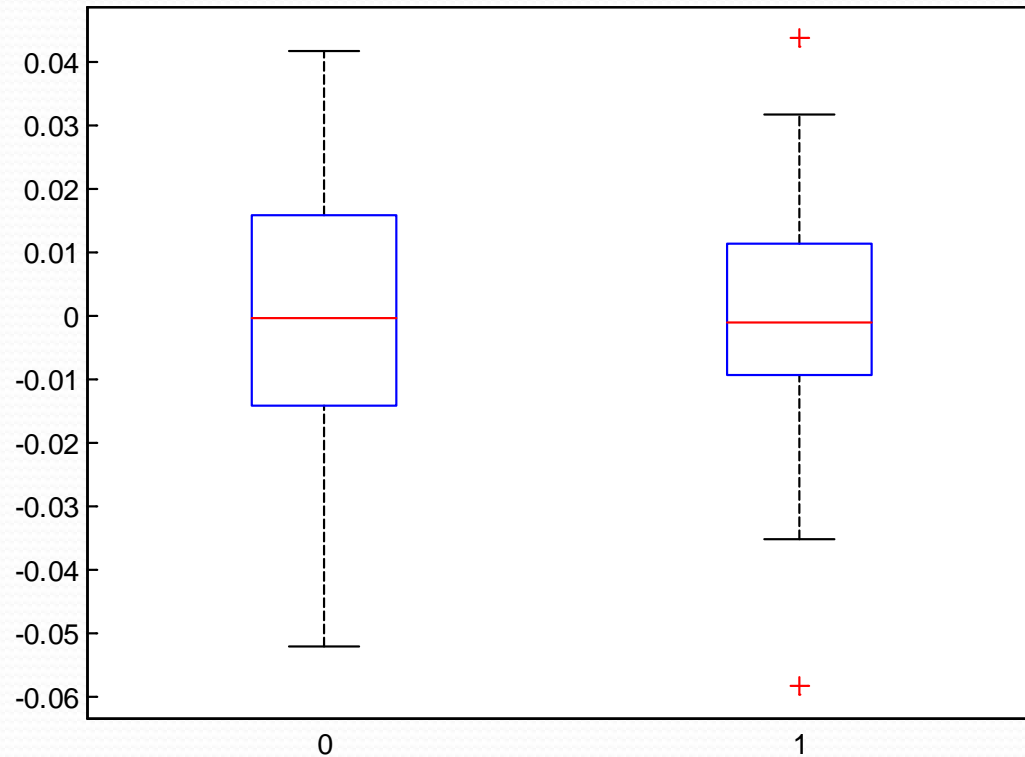
Residuals vs. Fitted Values



Residuals vs. Log Concentration



Residuals vs. Glucose



Inference for Linear Models

- The printed table of coefficients shows a test for whether each predictor can be eliminated. In cases where the predictors are correlated, this can cause loss of significance for both, because inclusion of either one gives good results.
- The F-statistic printed at the bottom is a test of whether the whole model is better than predicting the response only from the mean.

ANOVA

- The Analysis of Variance (ANOVA) is a method of dividing up the sum of squares into parts to test statistically if one of those parts is needed.
- $SST = SSR + SSE$
- We can compute the MS for these parts by dividing the SS by the df.
- Statistical tests come from ratios of mean squares using the F-distribution.
- The test for the whole model is MSR/MSE .

```
>> thllm2 = fitlm(th,'ldiameter ~ lconc * glucose')
```

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7642	0.044221	85.123	3.8116e-53
lconc	-0.059677	0.0039197	-15.225	8.5488e-20
glucose	-0.0078692	0.054559	-0.14423	0.88593
lconc:glucose	0.0064805	0.0048209	1.3442	0.18532

```
>> anova(thllm2,'components',3)
```

	SumSq	DF	MeanSq	F	pValue
lconc	0.10085	1	0.10085	231.8	8.5488e-20
glucose	9.0505e-06	1	9.0505e-06	0.020803	0.88593
lconc:glucose	0.00078615	1	0.00078615	1.807	0.18532
Error	0.020448	47	0.00043506		

This shows the increase in the error sum of squares when each variable individually is removed. For quantitative or binary variables, this gives the same results as the t-test of the coefficient.

'sstype' — Type of sum of squares

3 | 1 | 2 | h

Type of sum squares, specified as the comma-separated pair consisting of 'sstype' and the following:

- 1 — Type I sum of squares. The reduction in residual sum of squares obtained by adding that term to a fit that already includes the terms listed before it.
- 2 — Type II sum of squares. The reduction in residual sum of squares obtained by adding that term to a model consisting of all other terms that do not contain the term in question.
- 3 — Type III sum of squares. The reduction in residual sum of squares obtained by adding that term to a model containing all other terms.
- h — Hierarchical model. Similar to type 2, but with continuous as well as categorical factors used to determine the hierarchy of terms.

Components = 1 (sequential—each variable added in listed order)

Components = 2 (each variable added to model that includes all terms not involving that variable, with respect to factors not variables)

Components = 3 (each variable removed with all other variables kept)

Components = 'h' (each variable added to model including all terms not involving that variable) (default in 2015a for linear models anova)

Components = 1 (sequential—each variable added in listed order)
Components = 2 (each variable added to model including all terms not involving that variable, with respect to factors not variables)
Components = 3 (each variable removed with all other variables kept)
Components = 'h' (each variable added to model including all terms not involving that variable) (default in 2014a)

```
>> thllm2 = fitlm(th,'ldiameter ~ lconc * glucose')
```

Components = 1 (sequential)

lconc	ldiameter ~ 1 + lconc
	ldiameter ~ 1

glucose	ldiameter ~ lconc + glucose
	ldiameter ~ lconc

lconc:glucose	ldiameter ~ lconc + glucose + lconc:glucose
	ldiameter ~ lconc + glucose

Components = 1 (sequential—each variable added in listed order)
Components = 2 (each variable added to model including all terms not involving that variable, with respect to factors not variables)
Components = 3 (each variable removed with all other variables kept)
Components = 'h' (each variable added to model including all terms not involving that variable) (default in 2014a)

```
>> thllm2 = fitlm(th,'ldiameter ~ lconc * glucose')
```

Components = 'h' (hierarchical)

lconc	ldiameter ~ lconc + glucose
	ldiameter ~ glucose

glucose	ldiameter ~ lconc + glucose
	ldiameter ~ lconc

lconc:glucose	ldiameter ~ lconc + glucose + lconc:glucose
	ldiameter ~ lconc + glucose

Components = 1 (sequential—each variable added in listed order)
Components = 2 (each variable added to model including all terms not involving that variable, with respect to factors not variables)
Components = 3 (each variable removed with all other variables kept)
Components = 'h' (each variable added to model including all terms not involving that variable) (default in 2014a)

```
>> thllm2 = fitlm(th,'ldiameter ~ lconc * glucose')
```

Components = 2 or 3 (individual terms)

lconc	ldiameter ~ lconc + glucose + lconc:glucose
	ldiameter ~ glucose + lconc:glucose

glucose	ldiameter ~ lconc + glucose + lconc:glucose
	ldiameter ~ lconc + lconc:glucose

lconc:glucose	ldiameter ~ lconc + glucose + lconc:glucose
	ldiameter ~ lconc + glucose

```
>> anova(th1lm2,'components',1)
```

	SumSq	DF	MeanSq	F	pValue
lconc	0.24861	1	0.24861	571.44	6.0103e-28
glucose	0.050335	1	0.050335	115.7	2.8902e-14
lconc:glucose	0.00078615	1	0.00078615	1.807	0.18532
Error	0.020448	47	0.00043506		

```
>> anova(th1lm2,'components','h')
```

	SumSq	DF	MeanSq	F	pValue
lconc	0.25635	1	0.25635	589.24	3.0818e-28
glucose	0.050335	1	0.050335	115.7	2.8902e-14
lconc:glucose	0.00078615	1	0.00078615	1.807	0.18532
Error	0.020448	47	0.00043506		

Both sequential ANOVA and hierarchical ANOVA show clearly that adding the glucose variable is important, but adding the interaction of glucose and lconc is not. Thus it is proper to fit the model without the interaction, meaning that the slopes of the two lines can be considered to be the same. The last two tests are the same—only the first test is slightly different. The MATLAB default of 'h' is usually OK.

Specification of the Model

```
>> th = table(lconc, glucose, ldiameter)  #response last by default
>> fitlm(th)
```

Linear regression model:

ldiameter ~ 1 + lconc + glucose

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

A table is a way of putting all the variables in one object. You can also use dataset, but this is deprecated and may not exist in later MATLAB versions


```
>> fitlm(th)
```

Linear regression model:

ldiameter ~ 1 + lconc + glucose

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

```
>> fitlm(th,'ldiameter ~ lconc + glucose')
```

Linear regression model:

ldiameter ~ 1 + lconc + glucose

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

```
>> fitlm(th,'ldiameter ~ lconc + glucose')
```

Linear regression model:

ldiameter ~ 1 + lconc + glucose

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7161	0.026255	141.54	1.3865e-64
lconc	-0.055393	0.0023011	-24.073	1.9177e-28
glucose	0.06502	0.0060955	10.667	2.9322e-14

Number of observations: 51, Error degrees of freedom: 48

Root Mean Squared Error: 0.021

R-squared: 0.934, Adjusted R-Squared 0.931

F-statistic vs. constant model: 338, p-value = 5.24e-29

```
>> fitlm(th,'ldiameter ~ lconc*glucose')
```

Linear regression model:

ldiameter ~ 1 + lconc*glucose

Estimated Coefficients:

	Estimate	SE	tStat	pValue
(Intercept)	3.7642	0.044221	85.123	3.8116e-53
lconc	-0.059677	0.0039197	-15.225	8.5488e-20
glucose	-0.0078692	0.054559	-0.14423	0.88593
lconc:glucose	0.0064805	0.0048209	1.3442	0.18532

Number of observations: 51, Error degrees of freedom: 47

Root Mean Squared Error: 0.0209

R-squared: 0.936, Adjusted R-Squared 0.932

F-statistic vs. constant model: 230, p-value = 4.51e-28

Wilkinson Notation

- For details see the MATLAB help page for this.
- $Y \sim$ terms
- Variables are quantitative, factors are qualitative
- $A + B$ include both variables or factors
- $A:B$ include the A by B interaction or product
- $A*B$ include A and B and $A:B$
- A^2 when A is a variable, include the square
- Include the intercept (constant term) unless -1 is included in the formula