Curve fitting with least squares

Fitting functions to data

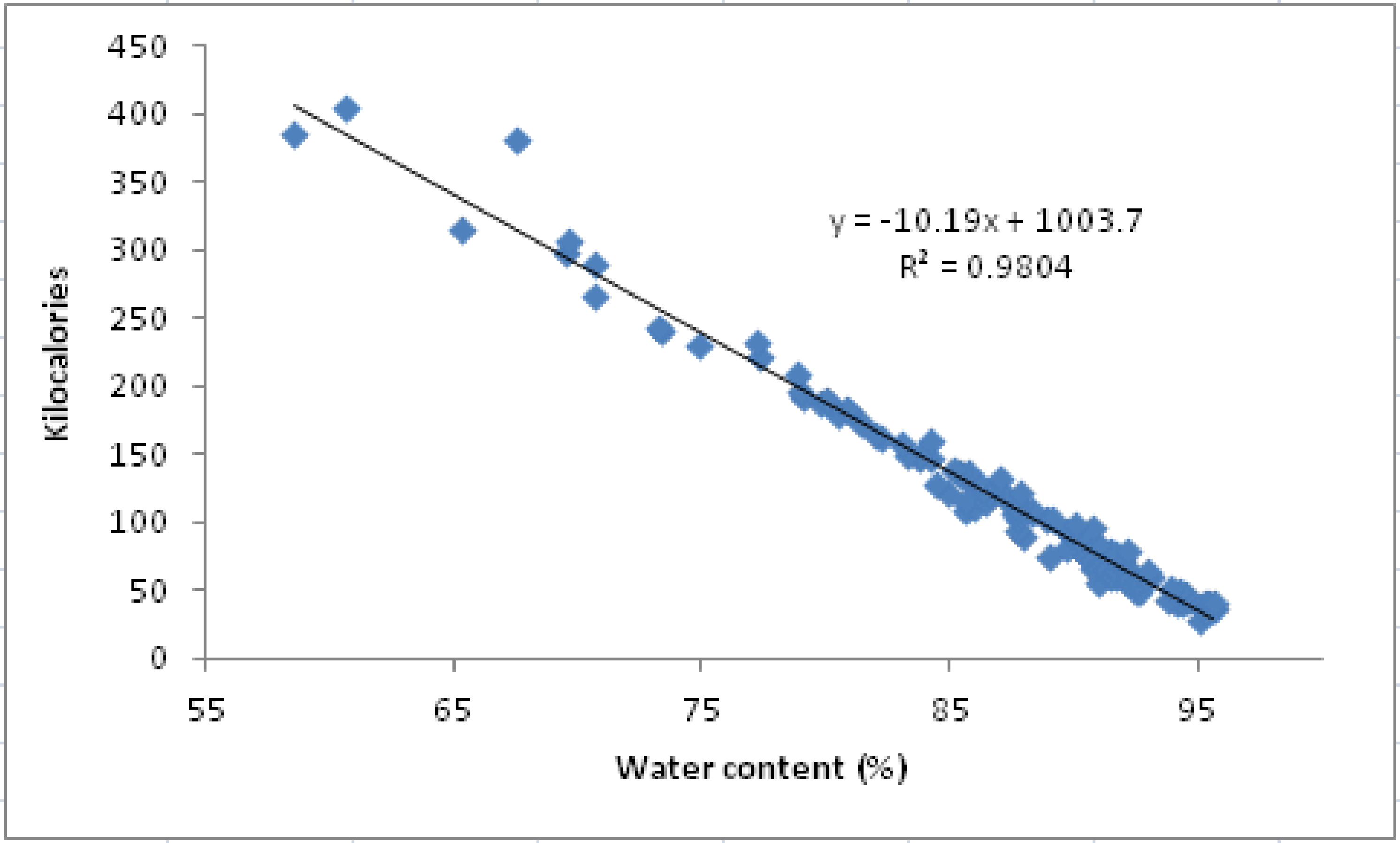
Fitting functions to data

- Common way to analyze data
- Two useful purposes
  - Assess the relationship between variables, obtain a predictive function
  - Obtain estimates of parameters
- We will focus today on “least squares” approaches
  - Least squares criterion: The line of best fit to the data minimizes the squared deviations between the data and the line

Simple linear regression

- Used to assess the straight-line relationship between two numeric variables
- Two variables
  - Independent, or predictor
  - Dependent, or response
- The independent is treated as the cause of change in the dependent
- Deviation from the line is treated as random variation, and only in the response variable

Regression of kilocalories on water content in various foods



Linear functions are easy to solve analytically

- There are equations for slope and intercept: $\hat{y} = a + b\,x$

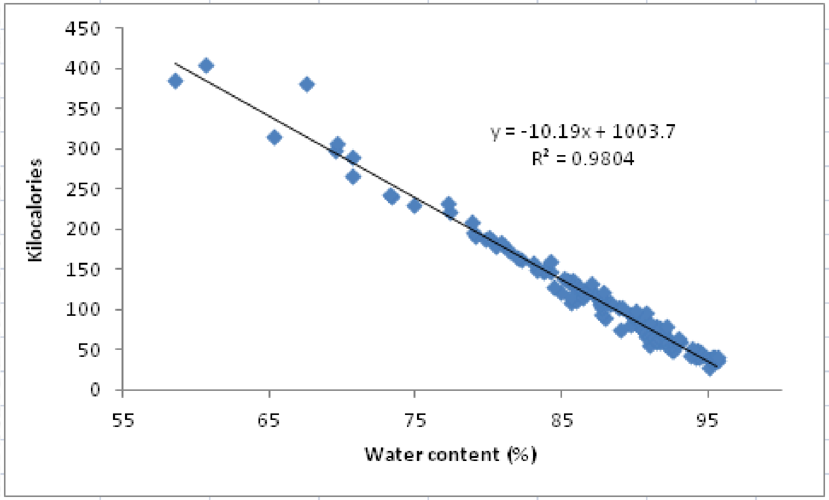
$$b = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sum (x_i - \bar{x})^2}$$

$$a = \bar{y} - b\,\bar{x}$$

- Equations also available for standard errors and 95% confidence intervals of the estimates
- But, some equations can't be so easily solved analytically
- Instead, we can use numerical approaches to fit the line, and obtain standard errors

Least squares

- Want the best fit line – how do we know we have it?
- Least squares criterion: the best fit line minimizes the squared deviations between the line and the data
  - Sum of squared deviations between the data and the line is the "residual sums of squares"
  - Sum of squared deviations of y data from y mean is the "total sums of squares"
  - Variation accounted for by the line is "explained" or "model sums of squares"



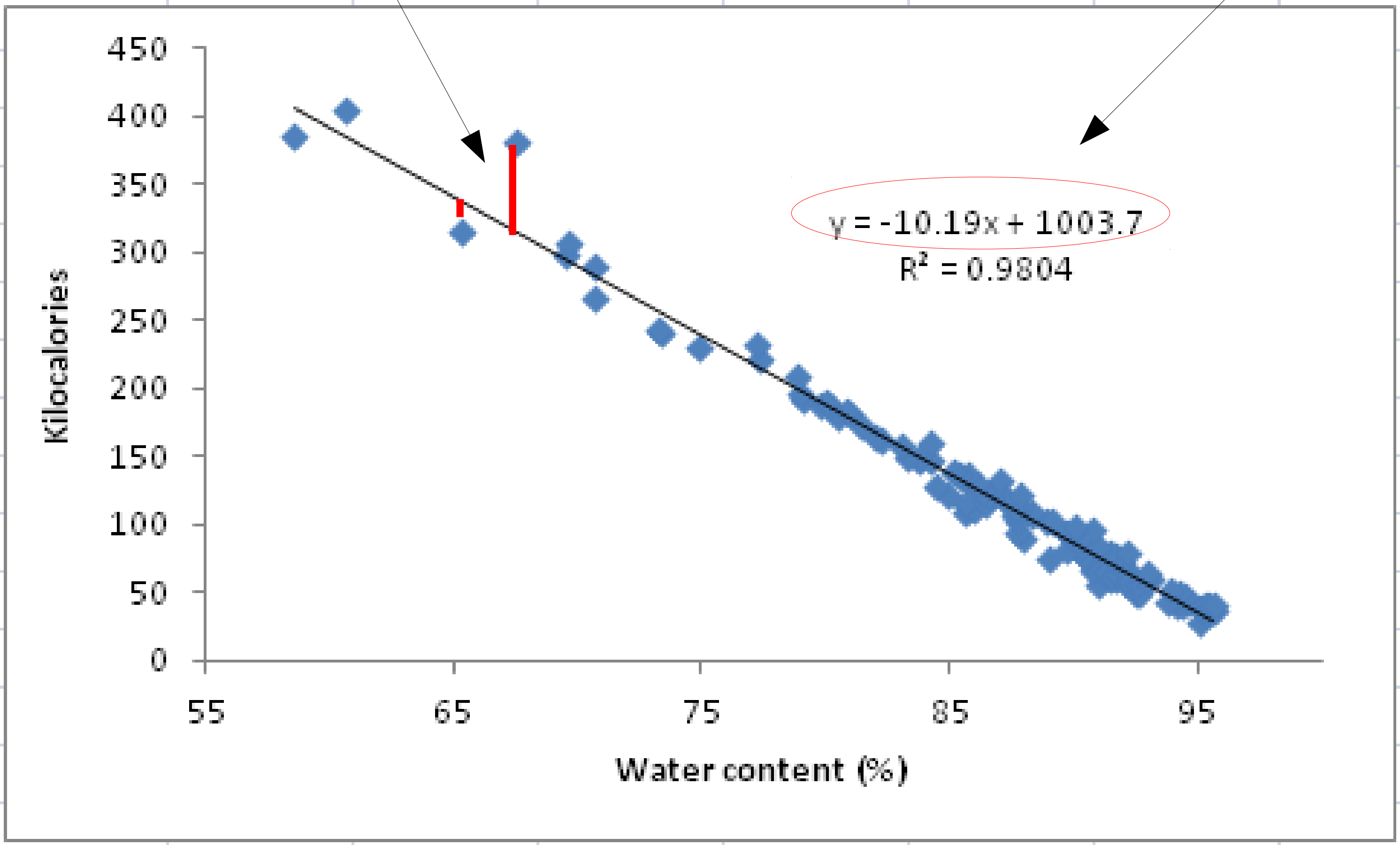
- r^2 = coefficient of determination
  - "Explained" sums of squares / total sums of squares
  - (Total SS – residual SS)/total SS

Residuals

Residual = observed – predicted value

Vertical differences only

Predicted value = average of y expected for a given value of x



Numerically fitting data to a function

- Start with a set of x and y data
- Use a function that predicts y from x, using any (reasonable) starting values for the unknown parameters (slope and intercept)
- Calculate the residuals, then square them
- Sum the squared residuals
- Use Solver to minimize the sum of squared residuals by changing the slope and intercept parameters

In Excel

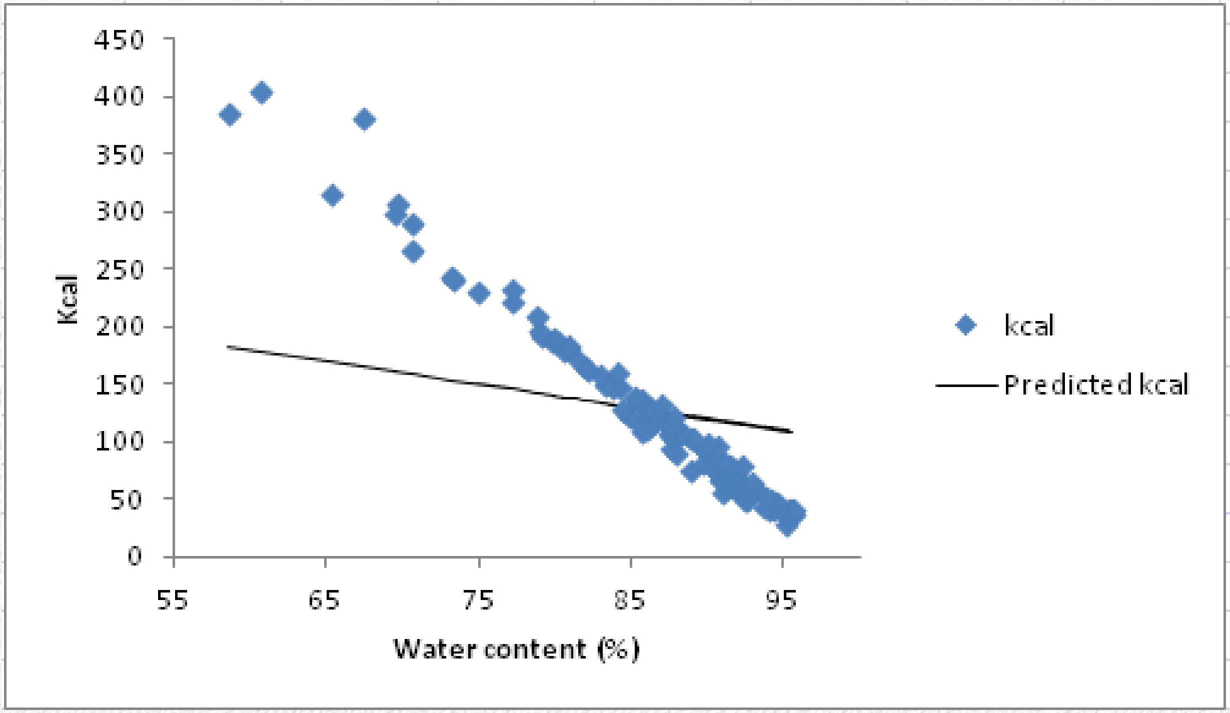
Predicted from straight line formula using initial parameters in B118 and B119

	A	B	C	D	E
1	food	water	kcal	Predicted kcal	Squared deviations
2	watercress	95.11	28	=B$118*B2+B$119	=(C2-D2)^2
3	pak-choi cabbage	95.32	34	109.36	5679.1
4	iceberg lettuce	95.64	36	108.72	5288.2
5	white gourd	95.54	36	108.92	5244.7
6	green leaf lettuce	95.07	38	109.86	5163.8
7	cucumber	95.23	40	109.54	4835.8
8	radish	95.27	41	109.46	4686.8
9	nopales	94.12	41	111.76	5007.0
113	plantains	65.28	316	169.44	21479.6
114	soybeans	67.5	381	165	46440.9
115	garlic	58.58	386	182.84	41274.4
116	prairie turnips	60.69	405	178.62	51248.4
117					
118	Slope	-2		Sum Sq.	452408.5
119	Intercept	300			
120					

Squared deviations between observed and predicted kcal's

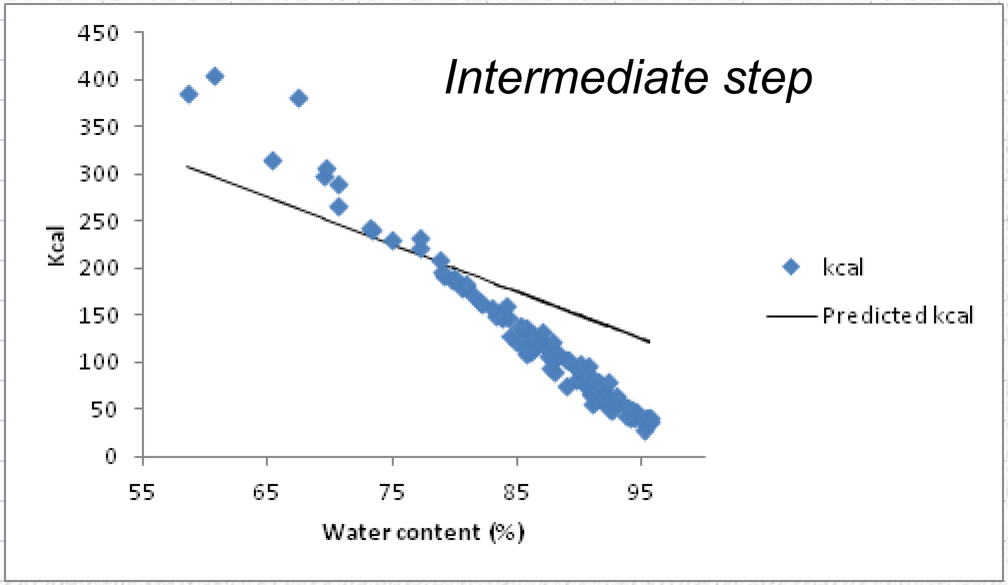
Sum of squared deviations – minimize with Solver

Close enough to start...

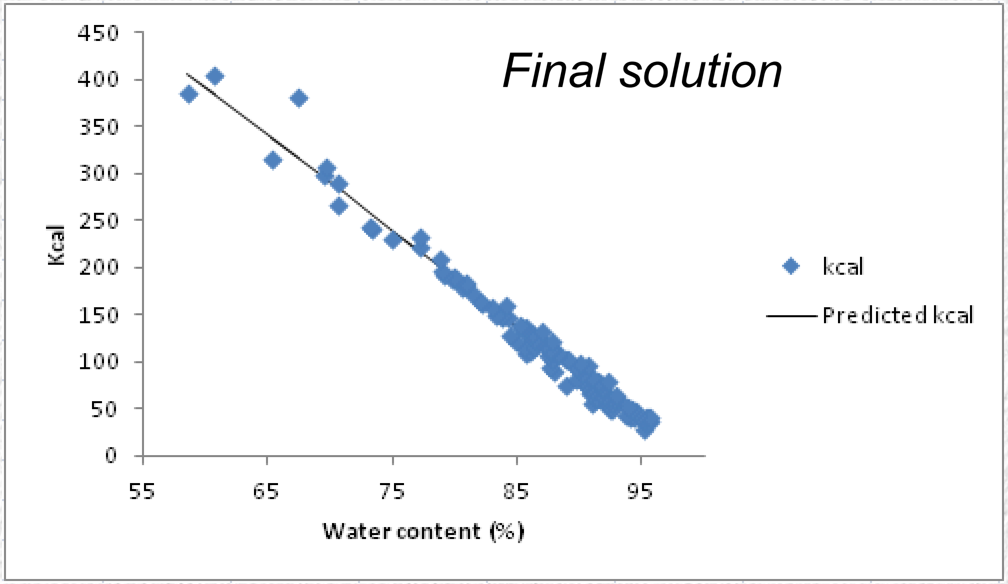


Slope = -2
Intercept = 300

As Solver changes slope and intercept...

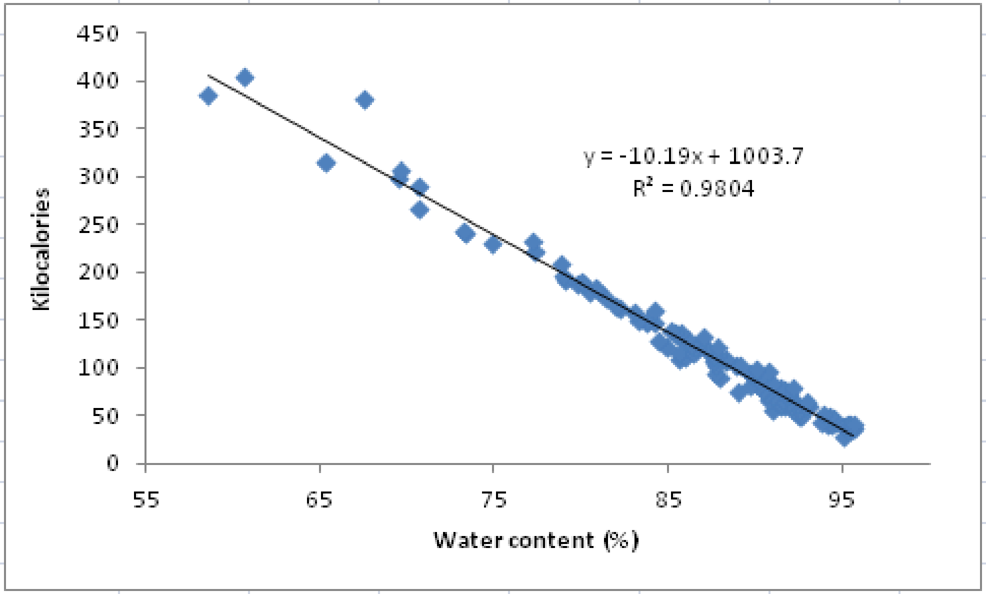


Slope = -5
Intercept = 600

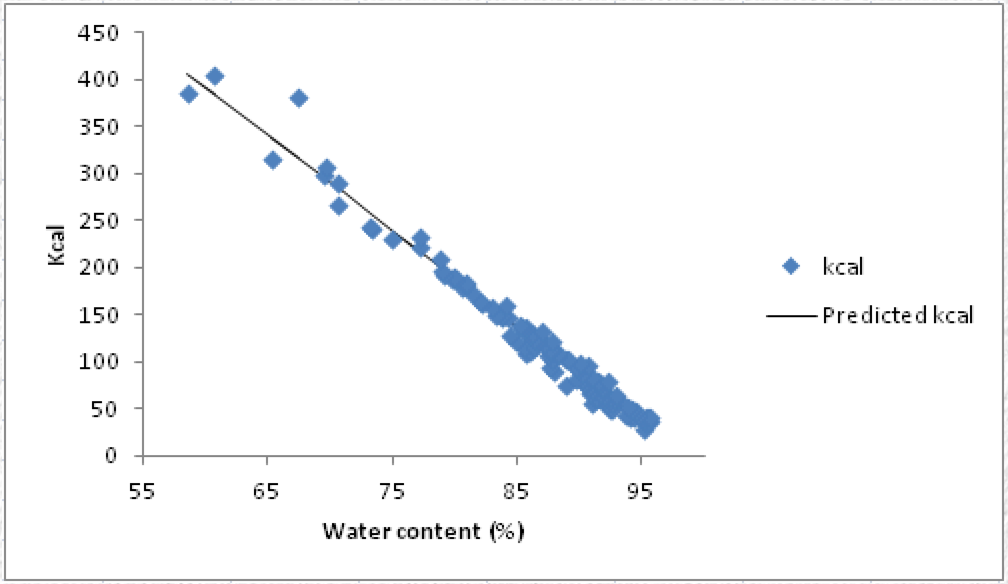


Slope = -10.19
Intercept = 1003.7

Match between analytical and numeric solutions



Very close agreement!



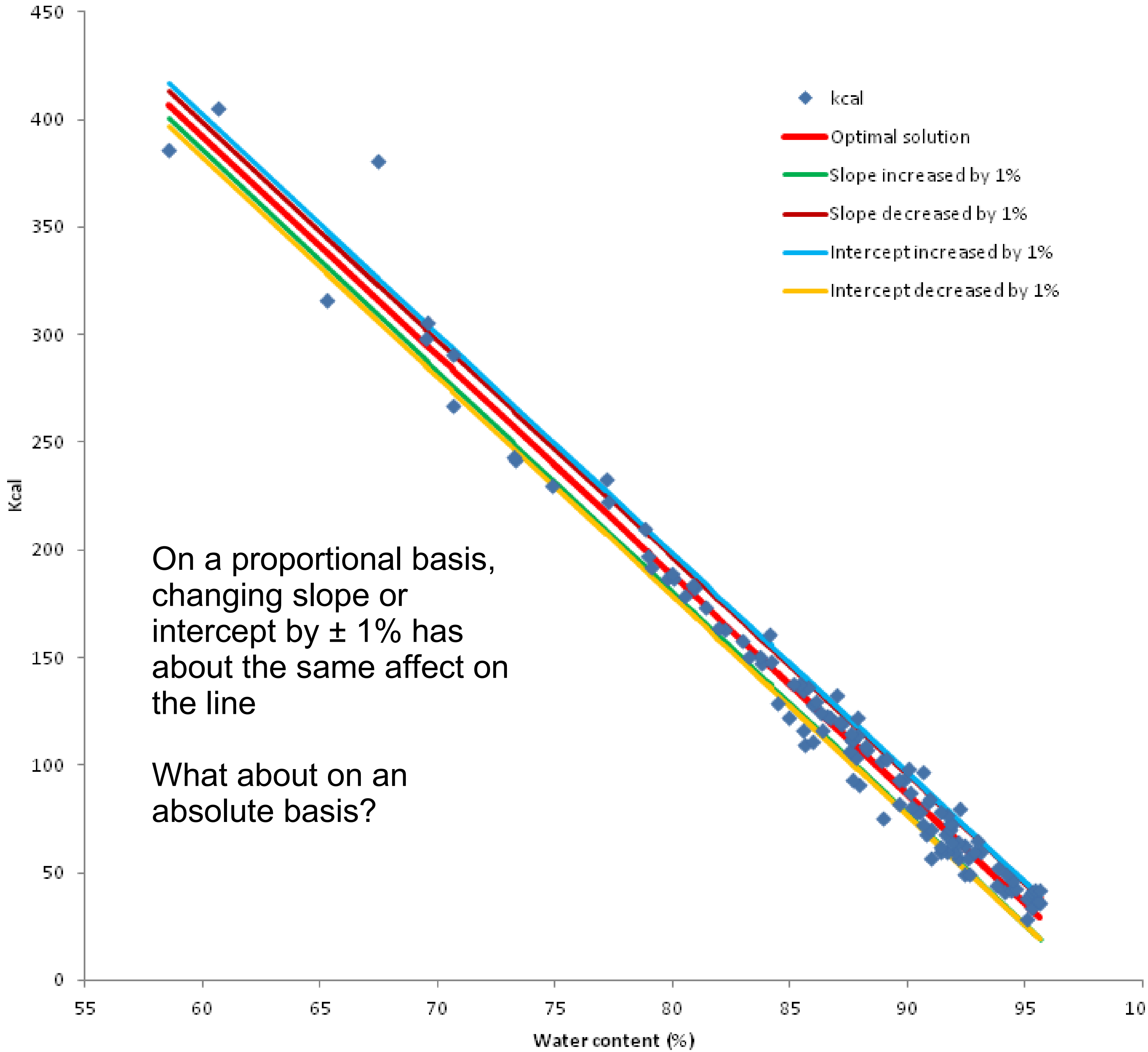
Slope = -10.19
Intercept = 1003.7

Problem: Solver does not provide standard errors

- Standard errors are measures of precision of estimates
  - A new set of data will give us different estimates
  - SE's used to measure how different we expect them to be
- They are also used for statistical hypothesis testing, and for calculating confidence intervals
- The slope and intercept estimates by themselves are not terribly useful without SE's
- We can estimate the SE's numerically with a little work, using “finite difference approximation”

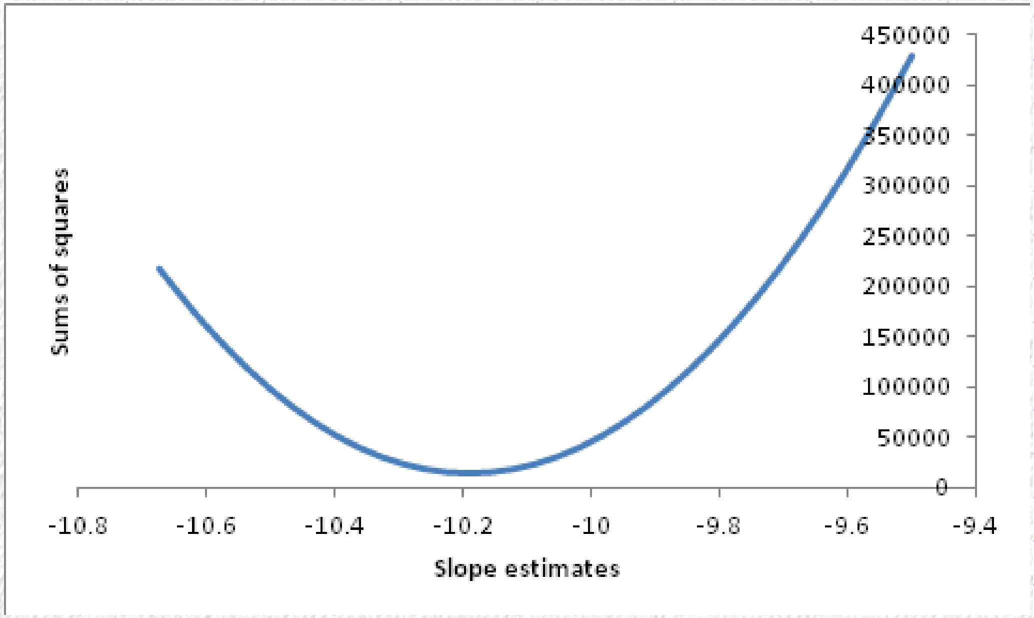
Basis for numerical SE estimation

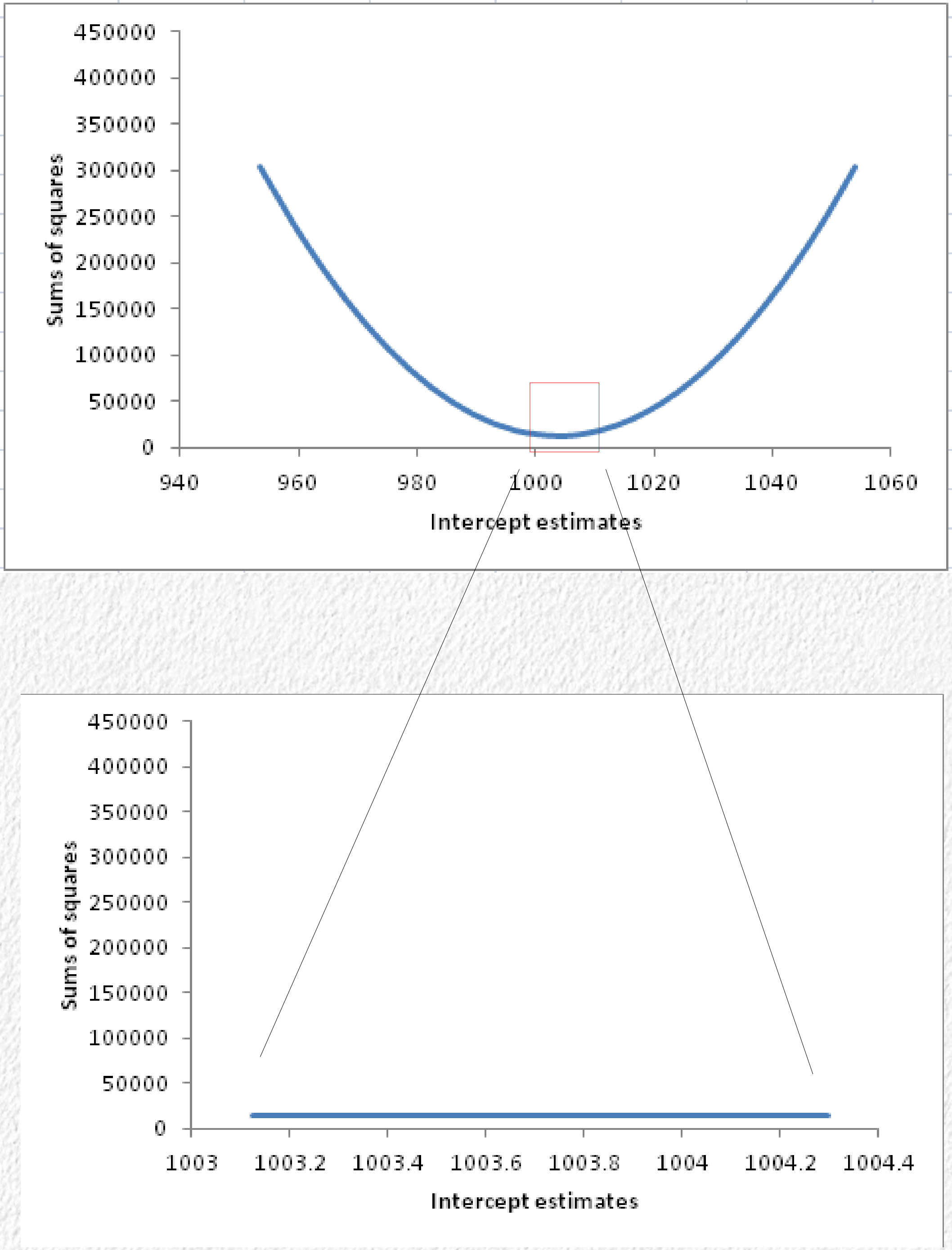
- The predicted values for the line are based on the estimated parameters
- If we vary one of the parameters at a time, we will change the predicted values by some amount
- Amount of change in predicted value per unit change in parameter value can be measured
  - Big changes in predicted value indicate better estimates
  - Small changes in predicted value indicate poorer estimates



On a proportional basis, changing slope or intercept by ± 1% has about the same affect on the line

What about on an absolute basis?





Fit of the line is much more sensitive to change in slope than change in intercept

So, the estimate of the slope will be more precise – the range of possible slopes that are consistent with the data is narrow

Finite difference approximation of SE

- Standard errors can be calculated using a matrix (P) of summed squared differences in predicted value per unit change in the estimates
- These approximate the first partial derivative of the line with respect to the estimate
  - Derivatives = slopes of lines tangent to a curve
  - Partial derivatives = derivative with respect to just one term, treating all others as constants
- The inverse of P can be used to estimate standard errors
- We can calculate P using tiny, finite changes to the parameters

The P matrix for the coefficients of a line

s = slope
i = intercept

$$P = \begin{bmatrix} \Sigma\left(\frac{\Delta f}{\Delta s}\right)^2 & \Sigma\frac{\Delta f}{\Delta s}\frac{\Delta f}{\Delta i} \\ \Sigma\frac{\Delta f}{\Delta s}\frac{\Delta f}{\Delta i} & \Sigma\left(\frac{\Delta f}{\Delta i}\right)^2 \end{bmatrix}$$

Cross products

Squared differences

- Derivatives used for continuous functions, instantaneous change
- We will use this as an approximation

Finite difference approximation of the partial derivatives

1) Start with the Solver estimates
2) Change the slope by a tiny amount
3) Calculate the change in the predicted values, divided by the change in the parameter
4) Return the slope to its Solver-estimated value
5) Repeat with the intercept
6) Calculate squares and cross-products, and sum them to estimate P

1. Predicted values from Solver estimates

	A	B	C	D
1	food	water	kcal	Predicted kcal
2	watercress	95.11	28	34.499
3	pak-choi cabbage	95.32	34	32.359
4	iceberg lettuce	95.64	36	29.098
5	white gourd	95.54	36	30.117
6	green leaf lettuce	95.07	38	34.907
7	cucumber	95.23	40	33.276
110	taro root	70.64	290	283.859
111	palm hearts	69.5	298	295.476
112	yam	69.6	306	294.457
113	plantains	65.28	316	338.479
114	soybeans	67.5	381	315.856
115	garlic	58.58	386	406.755
116	prairie turnips	60.69	405	385.253
117				
118	Slope	-10.1904		
119	Intercept	1003.709		
120				

Calculated as:

-10.1904 (water) + 1003.709

2. Change the slope – multiply slope by 1.000001

	food	water	kcal	Predicted kcal	
1	food	water	kcal	Predicted kcal	
2	watercress	95.11	28	34.489	
3	pak-choi cabbage	95.32	34	32.349	
4	iceberg lettuce	95.64	36	29.088	
5	white gourd	95.54	36	30.107	
6	green leaf lettuce	95.07	38	34.897	
7	cucumber	95.23	40	33.267	
110	taro root	70.64	290	283.851	
111	palm hearts	69.5	298	295.469	
112	yam	69.6	306	294.449	
113	plantains	65.28	316	338.472	
114	soybeans	67.5	381	315.850	
115	garlic	58.58	386	406.749	
116	prairie turnips	60.69	405	385.247	
117					
118	Slope	-10.1905			
119	Intercept	1003.709			
120					

Slight change in predicted values

Slope changed

Intercept kept constant

3. Change in predicted value divided by change in slope

D	E	F
Predicted kcal	Predicted kcal from Solver estimates	Differences (change in F)
34.489	34.4990905	0.010
32.349	32.3591036	0.010
29.088	29.0981712	0.010
30.107	30.1172126	0.010
34.897	34.906707	0.010
33.267	33.2762408	0.010
283.851	283.858514	0.007
295.469	295.475585	0.007
294.449	294.456544	0.007
338.472	338.479131	0.007
315.850	315.856413	0.007
406.749	406.754903	0.006
385.247	385.25313	0.006

÷

118	***Solver's solutions (best fit)***	
119	Slope	-10.1904
120	Int	1003.709
121		
122	***Altered solutions***	
123	Slope	-10.1905
124	Intercept	1003.709
125		
126	Change in slope	0.000102
127		

=

dY/ds
-95.11
-95.32
-95.64
-95.54
-95.07
-95.23
-70.64
-69.5
-69.6
-65.28
-67.5
-58.58
-60.69

4,5. Return slope to Solver value, repeat with intercept

Predicted kcal	Predicted kcal from Solver estimates	Differences (change in F)
34.500	34.4990905	-0.001
32.360	32.3591036	-0.001
29.099	29.0981712	-0.001
30.118	30.1172126	-0.001
34.908	34.906707	-0.001
33.277	33.2762408	-0.001
283.860	283.858514	-0.001
295.477	295.475585	-0.001
294.458	294.456544	-0.001
338.480	338.479131	-0.001
315.857	315.856413	-0.001
406.756	406.754903	-0.001
385.254	385.25313	-0.001

÷

L18	***Solver's solutions (best fit)***	
L19	Slope	-10.1904
L20	Int	1003.709
L21		
L22	***Altered solutions***	
L23	Slope	-10.1904
L24	Intercept	1003.71
L25		
L26	Change in intercept	-0.001

=

dY/di
1
1
1
1
1
1
1
1
1
1
1
1
1

Calculating sums of squares and cross products the “easy” way

- We just made a matrix with columns dY/ds and dY/di
- Want to square these and sum them for the main diagonal of P
- We want to multiply them together and sum them for the off-diagonals
- We can do this in one calculation using matrix multiplication – multiply the matrix by its transpose
- What's a transpose?

Transpose of a matrix

- A matrix is “transposed” by swapping the rows and columns

$$\boldsymbol{A}=\begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad \boldsymbol{A}'=\begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

$$\boldsymbol{A}'\times\boldsymbol{A}=\begin{bmatrix} a & c \\ b & d \end{bmatrix}\times\begin{bmatrix} a & b \\ c & d \end{bmatrix}=\begin{bmatrix} aa+cc & ab+cd \\ ba+dc & bb+dd \end{bmatrix}$$

Pre-multiplying a matrix by its transpose gives sums of squares and cross products

6. Calculate sums of squares and cross-products of dY/ds and dY/di

Font | Alignment | Number | Styles | Cells

{=MMULT(R2:EB3,N2:O116)}

M	N	O	P	Q	R	S	T	U	V
	dY/ds	dY/di							
	-95.11	1		dY/ds	-95.11	-95.32	-95.64	-95.54	-95.07
	-95.32	1		dY/di	1	1	1	1	1
	-95.64	1							
	-95.54	1							
	-95.07	1							
	-95.23	1			P Matrix, by finite difference approximation				
	-95.27	1							
	-94.12	1				Slope	Intercept		
	-95.43	1			Slope	870102.3	-9965.21		
	-94.39	1			Intercept	-9965.21	115		
	-95.64	1							
	-94.64	1							
	-93.79	1							

Array formula for matrix multiplication

dY/ds and dY/di, transposed

dY/ds and dY/di calculated by altering slopes and intercepts

Matrix multiplication of dY/ds, dY/di by transposed dY/ds, dY/di

Finally, calculate the standard errors

- To calculate the standard errors from the P matrix, we need to:
  - Invert the matrix
  - Multiply the square root of each of the values on the main diagonal by the standard error of Y
- This will give us both the standard errors we need
- What's a matrix inverse?

Matrix inverse

For a single number, a, the inverse is 1/a
a x 1/a = 1

$$\boldsymbol{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad \boldsymbol{A} \times \boldsymbol{A}^{-1} = \boldsymbol{I} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\boldsymbol{A} \times \boldsymbol{I} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = ?$$

We will let the computer solve inverses for us...

The inverse of our P matrix

{=MINVERSE(S10:T11)}

P Matrix, by finite difference approximation

	Slope	Intercept
Slope	870102.3	-9965.21
Intercept	-9965.21	115

Inverse of P matrix

	Slope	Intercept
Slope	0.000152	0.013175
Intercept	0.013175	1.150391

Standard errors

=SQRT(J122)*G120

F	G	H	I	J	K
	Squared deviations				
	36.0				
	2.7				
	47.6				
	40.7				
	9.6				
Sum Sq.	13641.3		Inverse of the P matrix		
SE(y)	10.98725282				
				Slope	Intercept
SE(s)	0.135480164		Slope	0.000152	0.013175
SE(i)	11.7845214		Intercept	0.013175	1.150391

Standard error of y is:

$$SE(Y) = \sqrt{\frac{SSY}{n-2}} = 10.987$$

SE of slope is:

$$SE(s) = \sqrt{P^{-1}{}_{11}}\, SE(Y) = \sqrt{0.000152}\,(10.987)$$

SE of intercept is:

$$SE(s) = \sqrt{P^{-1}{}_{22}}\, SE(Y) = \sqrt{1.150391}\,(10.987)$$

Tests of significance for coefficients

- The coefficient divided by its standard error can be tested as a t-value
- Use the error degrees of freedom for the model
- The test is whether the coefficient is equal to 0
  - If you fail to reject this, the coefficient isn't significant, isn't needed in the model
  - If you reject this, the coefficient is significant, is needed in the model

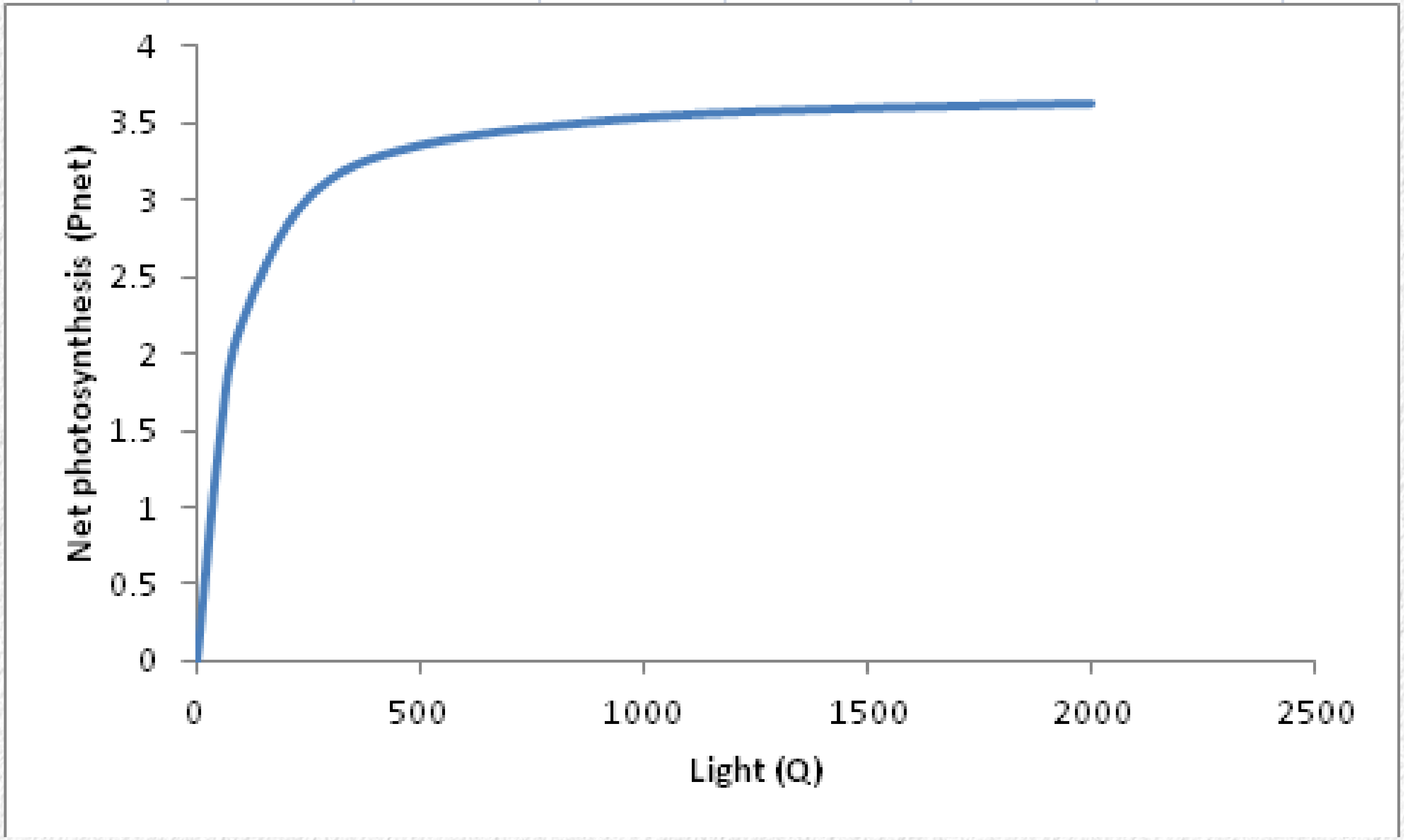
Significance tests

Coefficient	Estimate	SE	T	df	p
s	-10.19	0.1355	-10.19/0.1355 = -75.22	113	2.3×10^{-98}
i	1003.71	11.78	1003.71/11.78 = 85.17	113	2.4×10^{-104}

A trickier problem

- Sometimes what we can measure and what we want to know are two different things
- If we know how the quantity we want to know is related to the things we can measure, we can:
  - Use a function that shows the relationship
  - Fit the function to the data we can measure
  - Use the parameters from the best fit line as estimates of the quantities we are interested in
- Example: photosynthesis data

Net photosynthesis as a function of light intensity



A model of photosynthesis

- A mechanistic model that explains the relationship between light intensity and net photosynthesis is:

$$P_{net} = \frac{\Phi Q + P_{marea} - \sqrt{(\Phi Q + P_{marea})^2 - 4\theta \Phi Q P_{marea}}}{2\theta}$$

- By fitting this function to the data, it's possible to get estimates of each of the parameters
- The parameters have biological interpretations

To be estimated

Φ = Phi = Maximum quantum yield (CO_2 molecules fixed per photon)

P_{marea} = Maximum area-based rate of net photosynthesis (CO_2 per m^2 per s)

θ = Theta = Convexity of the curve (dimensionless – adjusts curve shape)

Known (the data)

Q = Light intensity (predictor variable, set by photosynthesis system)

P_{net} = Net photosynthesis (response variable, measured by photosynthesis system)

In Excel - setup

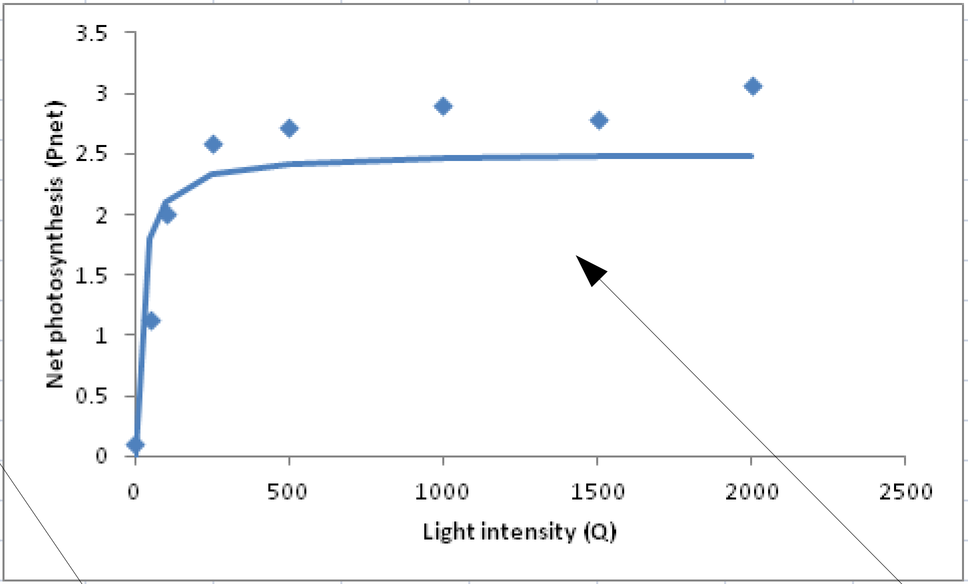
The data

Predicted Pnet from equation

C2 =(Phi*A2+Pmarea-((Phi*A2+Pmarea)^2-4*Theta*Phi*A2*Pmarea)^0.5)/(2*Theta)

	A	B	C	D
1	Q	Pnet	Predicted	(Obs - Predicted)^2
2	0	0.109075	0	0.011897
3	50	1.130192	1.79564	0.44282
4	100	2.000815	2.106496	0.011168
5	250	2.583246	2.332056	0.063096
6	500	2.719568	2.414259	0.093214
7	1000	2.899914	2.456689	0.196449
8	1500	2.788039	2.471028	0.100496
9	2000	3.059445	2.478235	0.337805
10				
11	Theta	0.3	SumSq	1.256947
12	Phi	0.1		
13	Pmarea	2.5		

Squared deviations

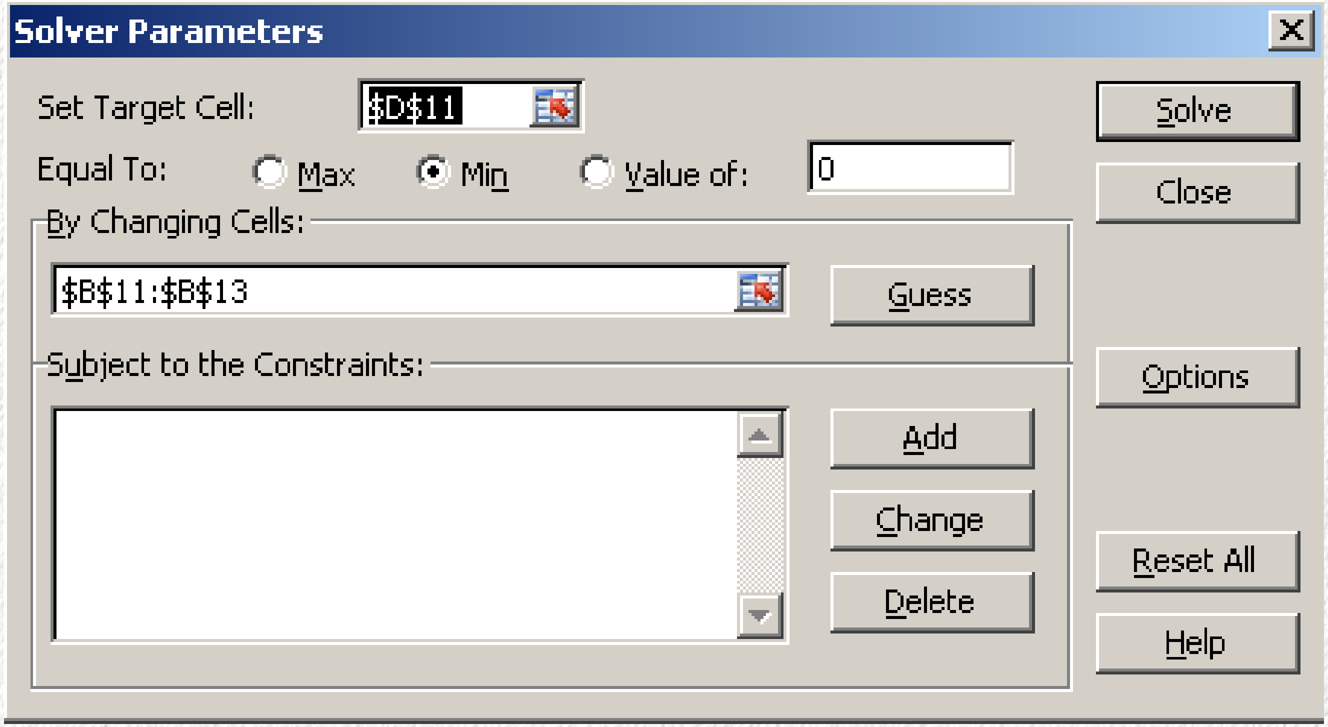


Parameters to estimate

Sum of squared deviations

Graph of observed and predicted (based on starting values of parameters)

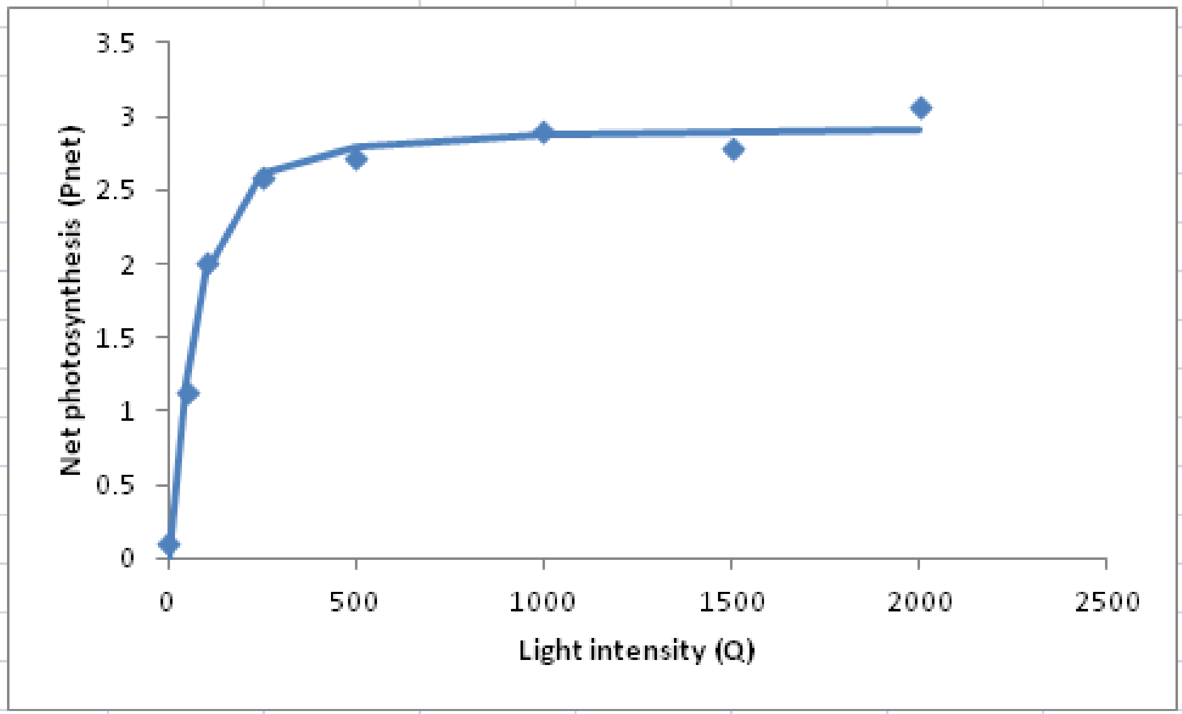
Solver settings



Solver's solution

D11 f_x =SUM(D2:D9)

	A	B	C	D
1	Q	Pnet	Predicted	(Obs - Predicted)^2
2	0	0.109075	0	0.011897
3	50	1.130192	1.182138	0.002698
4	100	2.000815	1.933934	0.004473
5	250	2.583246	2.606913	0.00056
6	500	2.719568	2.793662	0.00549
7	1000	2.899914	2.872733	0.000739
8	1500	2.788039	2.896999	0.011872
9	2000	3.059445	2.908752	0.022708
10				
11	Theta	0.797061	SumSq	0.060438
12	Phi	0.026865		
13	Pmarea	2.942538		



Standard errors

- We can use the same methods we used for regression
- Only difference is now we have three parameters

Calculate deltas

Predicted from modified parameters

	A	B	C	D	E	F	G	H	I
1	Q	Pnet	Predicted	(Obs - Predicted)^2		Solver predicted	dY/dTheta	dY/dPhi	dY/dPmarea
2	0	0.109075	0	0.011897		0	0	0	0
3	50	1.130192	1.182138	0.002698		1.1821377	0.581955041	36.65509338	0.067087787
4	100	2.000815	1.933935	0.004473		1.9339337	1.468962768	39.61390109	0.295566499
5	250	2.583246	2.606915	0.00056		2.6069128	1.234966444	15.24741928	0.746734242
6	500	2.719568	2.793665	0.00549		2.7936621	0.654663839	6.244039967	0.892398646
7	1000	2.899914	2.872736	0.000739		2.8727327	0.327123371	2.767013381	0.951014787
8	1500	2.788039	2.897002	0.011873		2.8969987	0.217304158	1.768694101	0.968375841
9	2000	3.059445	2.908755	0.022707		2.9087522	0.162598782	1.29859205	0.976662128
10									
11	Theta	0.797061	SumSq	0.060438					
12	Phi	0.026865							
13	Pmarea	2.942541							
14									
15	Theta	0.797061	1						
16	Phi	0.026865	1						
17	Pmarea	2.942538	1.000001						

Predictions from Solver's estimates

Change in predicted value divided by change in parameter value

Calculate P (matrix, not P_{net})

G	H	I	J	K	L	M	N	O
dY/dTheta	dY/dPhi	dY/dPmarea						
0	0	0						
0.581955041	36.65509338	0.067087787						
1.468962768	39.61390109	0.295566499						
1.234966444	15.24741928	0.746734242						
0.654663839	6.244039967	0.892398646						
0.327123371	2.767013381	0.951014787						
0.217304158	1.768694101	0.968375841						
0.162598782	1.29859205	0.976662128						
Deltas transposed								
dY/dTheta	0	0.581955041	1.468963	1.234966	0.654664	0.327123	0.217304	0.162599
dY/dPhi	0	36.65509338	39.6139	15.24742	6.24404	2.767013	1.768694	1.298592
dY/dPmarea	0	0.067087787	0.295566	0.746734	0.892399	0.951015	0.968376	0.976662
P matrix								
4.6	103.9	2.7						
103.9	3196.8	36.7						
2.7	36.7	4.2						

Invert P, calculate standard errors

P inverse				
1.45	-0.04	-0.56		
-0.04	0.00	0.01		
-0.56	0.01	0.48		
SE(Y) - 8 observations, 3 fitted parameters = 5 df				
0.109943554				
Parameter	Estimate	SE	t	p
Theta	0.797	0.132	6.0	0.002
Phi	0.027	0.004	6.3	0.001
Pmarea	2.943	0.076	38.8	2.14E-07