

Least squares estimation of parameters for various models of enzyme inhibition.

This exercise has two goals: (1) to give you an understanding of how least squares estimation of parameters works in practice, and (2) to increase your understanding of the differences in kinetic behavior associated with “competitive”, “uncompetitive” and “mixed type” inhibition of enzyme activity. The theoretical background for least squares estimation of parameters is given in a separate handout entitled “Handout on least-squares estimation of parameters” (<http://www.haverford.edu/chem/scarrow/leastsquares/theory.pdf>). For typographical reasons, I made two symbol changes relative to that handout:

- The \mathbf{V} vector ($n_\theta \times 1$ matrix) of the least squares handout is renamed the \mathbf{U} matrix to avoid confusion with the rates of the enzyme reaction (traditionally denoted as V by biochemists), and also because, in some fonts, \mathbf{V} is hard to distinguish from \mathbf{Y} on the computer screen.
- The parameter vector θ is renamed \mathbf{P} (for parameter) to avoid having to use Greek characters in the EXCEL spreadsheet (also, n_θ is renamed n_p).

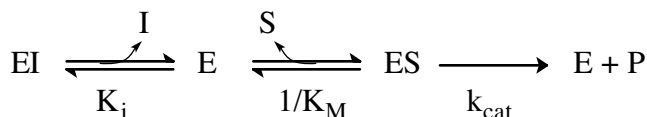
The theory of enzyme inhibition types is given in Biochemistry text books, and a very condensed version is given below.

Theory behind different types of enzyme inhibition.

An enzyme inhibitor I is said to be **competitive** with the substrate S (that is, the normal reactant for the catalyzed reaction) if the rate of the catalyzed reaction (denoted by the symbol V) obeys the following equation (it is assumed that the enzyme concentration is the same for all measurements of V):

$$V = \frac{V_{\max}}{1 + \frac{K_m}{[S]}\left(1 + \frac{[I]}{K_i}\right)} \quad (\text{comp: 1})$$

A very simple reaction mechanism that explains competitive inhibition is as follows, where $k_{\text{cat}} = V_{\max} / [E]_{\text{total}}$ and $[E]_{\text{total}}$ is the concentration of enzyme in all its forms (E , ES , EI , etc.).



The inhibitor and substrate molecules compete for binding to the enzyme - they don't both bind at the same time. A different equation (note the addition of parentheses) would be obtained if the inhibitor binds (with the same affinity) to the ES complex as well as the substrate-free E --in this case, the inhibitor is said to be **non-competitive**:

$$V = \frac{V_{\max}}{\left(1 + \frac{K_m}{[S]}\right)\left(1 + \frac{[I]}{K_i}\right)} \quad (\text{non-comp:2})$$

A third possibility is that the inhibitor binds to the ES complex, but not to the enzyme without substrate. This is said to be **uncompetitive** inhibition, and is characterized by the following equation:

$$V = \frac{V_{\max}}{\left(1 + \frac{[I]}{K_i^u}\right) + \frac{K_m}{[S]}} \quad (\text{uncomp:3})$$

In a fourth, more general case of a **mixed** inhibitor, inhibitor binds to E and to ES, but with different affinities. Then

$$V = \frac{V_{\max}}{\left(1 + \frac{[I]}{K_i^u}\right) + \frac{K_m}{[S]}\left(1 + \frac{[I]}{K_i^c}\right)} \quad (\text{mixed: 4})$$

where K_i^u and K_i^c are the uncompetitive and competitive inhibition constants (dissociation constants for $ESI = ES + I$ and $EI = E + I$, respectively).

In all of these equations, V_{\max} is the maximum rate with lots of substrate and no inhibitor. K_m is the concentration of substrate at which the rate is half-maximal in the absence of inhibitor.

Kinetic data for rates in the absence of inhibitor can be linearized by plotting $1/V$ vs. $1/[S]$. This forms the basis for the familiar Lineweaver-Burke plots of enzyme reaction rates. From a mathematical point of view, $\frac{1}{V} = \frac{1}{V_{\max}} + \frac{K_m}{V_{\max}[S]}$ is a linear equation because it is of the form $y = P_1x_1 + P_2x_2$, where P_1 and P_2 are independent parameters which are the same for every data point (but with initially unknown values), and x_1 and x_2 are experimentally adjustable independent variables ($x_2 = 1/[S]$) or known constants ($x_1 = 1$).

Similarly, equation (1) can be linearized as:

$$\frac{1}{V} = \frac{1}{V_{\max}} + \frac{K_m}{V_{\max}}\left(\frac{1}{[S]}\right) + \frac{K_m}{V_{\max}K_i}\left(\frac{[I]}{[S]}\right) \quad (\text{comp: 1a})$$

That is

$$y = P_1x_1 + P_2x_2 + P_3x_3 \quad (\text{comp: 1b})$$

where $y = \frac{1}{V}$, $P_1 = \frac{1}{V_{\max}}$, $P_2 = \frac{K_m}{V_{\max}}$, $P_3 = \frac{K_m}{V_{\max}K_i}$, $x_1 = 1$, $x_2 = \frac{1}{[S]}$, and $x_3 = \frac{[I]}{[S]}$.

Equation (3) can be linearized as:

$$\frac{1}{V} = \frac{1}{V_{\max}} + \frac{K_m}{V_{\max}}\left(\frac{1}{[S]}\right) + \frac{1}{V_{\max}K_i}[I] \quad (\text{uncomp: 3a})$$

That is

$$y = P_1x_1 + P_2x_2 + P_3x_3 \quad (\text{uncomp: 3b})$$

where y , P_1 , P_2 , x_1 , and x_2 are as in equation (1b), $P_3 = \frac{1}{V_{\max} K_i}$, and $x_3 = [I]$

Equation (4) can be linearized as:

$$\frac{1}{V} = \frac{1}{V_{\max}} + \frac{K_m}{V_{\max}} \left(\frac{1}{[S]} \right) + \frac{K_m}{V_{\max} K_i^c} \left(\frac{[I]}{[S]} \right) + \frac{1}{V_{\max} K_i^u} ([I]) \quad (\text{mixed: 4a})$$

That is

$$y = P_1x_1 + P_2x_2 + P_3x_3 + P_4x_4 \quad (\text{mixed: 4b})$$

where y , P_1 , P_2 , x_1 , x_2 , and x_3 are as in equation (1b),

$$P_3 = \frac{K_m}{V_{\max} K_i^c} \text{ (also essentially the same as eq. 1b) , } P_4 = \frac{1}{V_{\max} K_i^u} \text{ , and } x_4 = [I].$$

Equation (2) cannot be linearized. At first glance, it might appear that equations 4a and 4b would also work, with the caveat that $K_i^c = K_i^u = K_i$ of equation 2. However, one of the conditions of linearization is that the parameters be mathematically independent, and this is not the case with equation 4b if $K_i^c = K_i^u$ (because $P_4 = P_1P_3/P_2$). In order to fully investigate the possibility of non-competitive inhibition, it will be necessary to use non-linear least squares refinement. That is the topic of exercise 5. But we will start by using linear least squares techniques to determine if a sample data set is fit better by assuming competitive, uncompetitive or mixed inhibition.

The remainder of this document consists of

- [Exercise 1\) Unweighted linear least squares assuming competitive inhibition](#)
- [Exercise 2\) Weighted linear least squares assuming competitive inhibition](#)
- [Exercise 3\) Weighted linear least squares assuming uncompetitive inhibition](#)
- [Exercise 4\) Weighted linear least squares assuming mixed inhibition](#)
- [Exercise 5\) Weighted non-linear least squares assuming various models of inhibition](#)

Also available on the internet are the following resources:

- A report form for summarizing the results from these exercises may be found on the web at <http://www.haverford.edu/chem/scarrows/leastsquares/reportform.pdf>.
- Partial answers (to check if you are on the right track) can be found at <http://www.haverford.edu/chem/scarrows/leastsquares/partialanswers.pdf>.

This document is currently available for download at <http://www.haverford.edu/chem/scarrows/leastsquares/exercise.pdf>.

Exercise 1: Given Dataset A, calculate the least squares estimates of P_1 , P_2 and P_3 assuming competitive inhibition using equation 1a. Then calculate V_{\max} , K_m and K_i .

The following table is of measured V values as a function of $[S]$ and $[I]$ concentrations, which have units of mM. The V_{obs} are in units of nmol of reaction catalyzed / minute. The data from this table are used for Exercises 1 through 4.

Dataset A: (download at <http://www.haverford.edu/chem/scarro/leastsquares/enzymeA.xls>)

	no inhibitor	[I] = 0.3	[I] = 1.0	[I] = 3.0	[I] = 10.0
[S] = 0.1	$V_{\text{obs}} = 0.9$	0.8	0.6	0.4	0.2
[S] = 0.2	1.9	1.7	1.5	0.5	0.1
[S] = 0.3	2.3	2.1	1.6	1.2	0.5
[S] = 0.4	2.2	2.5	1.7	1.5	0.8
[S] = 0.6	2.8	2.8	2.4	1.6	1.1
[S] = 1.0	3.5	3.4	2.9	2.5	1.2

An EXCEL spreadsheet containing the data in the table above is available from the URL listed above. The data has already been formatted as suggested in step 1 below, and also has various sections color-coded to make it easy to find the cells into which the various matrices will be placed. However, no matrix names have been defined yet; you will have to figure out how to do this yourself (see step 3).

1) In EXCEL, make a data table showing dataset A. Use one row for each measured V_{obs} , starting with data in row 12. Put $[I]$, $[S]$ and V_{obs} in columns A - C. In case you want to later plot your data, it is useful to arrange the data points such that the first 6 rows (rows 12-17) have $[I]=0$, the second 6 rows have $[I]=0.3$, etc. In carrying out your calculations, I suggest you keep the units of mM and nmol/min rather than converting to other units. Later, you'll need to figure out the correct units for all your results using equation 1 or 2.

2) Leave columns D and E blank (for use later). Calculate the \mathbf{y} vector ($1/V_{\text{obs}}$ values) in column F. This will be a 30×1 matrix.

3) Leave another column blank (for later). Enter/Calculate the \mathbf{X} matrix in columns H - J (column H will be all 1's, column I will be the appropriate formula (filled down) for $1/[S]$, and column J will be $[I]/[S]$).

Working with matrices in EXCEL is a bit tricky. For each matrix you use, I suggest you do the following to prevent confusion and help you follow what you are doing:

- Label each matrix with its name using a cell above or to the left of the matrix.
- Draw an Outline Border around each matrix (if you are working on a color monitor you could try making each matrix a different color instead).
- While you have the matrix selected for outlining, choose **Name>Define** from the **Insert** menu (or **Formula** menu if you are using EXCEL version 4). Give the matrix a name (\mathbf{y} and \mathbf{X} work fine as names). The names can then be used in subsequent formulae.

4) The procedure for entering a matrix formula is described in detail for the \mathbf{M} matrix ($\mathbf{M} = \mathbf{X}'\mathbf{X}$). Select cells K2:M4. Then type in the formula:

$$=MMULT(TRANSPOSE(X),X)$$

Then hold down the control and shift keys (or the command key on a Mac) and hit enter. The formula you just typed in will be enclosed in brackets (meaning it is a matrix formula, which can only be entered with control-shift-enter or (Mac) command-enter). $MMULT(\mathbf{a},\mathbf{b})$ returns the matrix product \mathbf{ab} , while $TRANSPOSE(\mathbf{a})$ returns the transpose. The other matrix functions provided by EXCEL are $MINVERSE(\mathbf{a})=\mathbf{a}^{-1}$, and $MDETERM(\mathbf{a})$ which gives the determinant of \mathbf{a} . (You learn about determinants in linear algebra courses, but we won't use them.)

5) In cells N2:P4, calculate \mathbf{M}^{-1} (call this 3×3 matrix \mathbf{invM}). Use the matrix formula:

$$=MINVERSE(M)$$

6) In cells Q2:Q4, calculate $\mathbf{U} = \mathbf{X}'\mathbf{y}$. (This was called \mathbf{V} in the handout on least-squares theory).

7) In cells C2:C4, calculate the parameters $\mathbf{P} = \mathbf{M}^{-1}\mathbf{U}$. (Called θ in the theory handout)

8) In cells L12:L41, calculate $\mathbf{y}_{\text{calc}} = \mathbf{X} \mathbf{P}$.

9) In cells M12:M41, calculate $\mathbf{y}_{\text{diff}} = \mathbf{y}_{\text{calc}} - \mathbf{y}$.

10) In cells E12:E41, calculate $\mathbf{V}_{\text{calc}} (=1/\mathbf{y}_{\text{calc}})$.

11) In cell B2, calculate "Resid" (called R in the least-squares notes) as the matrix equation $=SUM((YDIFF)^2)$. (This must be entered as a matrix equation (control-shift-enter), even though it occupies a single cell). The size of the residual depends, among other things, on how many data points there are and the magnitude of the data. One can compare "Resid" to "Rdata" $=SUM((Y)^2)$. Calculate Rdata in cell B1. The ratio Resid/Rdata is known as a "crystallographic R factor". Calculate this in cell B3.

12) In cell B4, calculate the GOF $=\sqrt{\text{Resid}/(n_{\text{data}} - n_p)}$. $SQRT()$ is the square root function in EXCEL. For *unweighted* least squares, the GOF is an estimation of the magnitude of the measurement error associated with individual y ($1/V$) values, assuming that the standard deviation for each y measurement is identical. (You may use cells A1:A4 for labels for Resid, Rdata, Xtal R factor, and GOF).

13) In cells E2:G4, calculate the variance-covariance matrix ($\mathbf{S} = \frac{R}{n_{\text{data}} - n_p} \mathbf{M}^{-1} = \text{GOF}^2 \mathbf{M}^{-1}$). To do this, use ordinary multiplication within the matrix equation $\{=(\text{GOF})^2 * \text{INVM}\}$.

14) In cells D2:D4, calculate the e.s.d.'s of the \mathbf{P} values (" $\mathbf{esd_P}$ ") as the square roots of the diagonal elements of \mathbf{S} . (Unfortunately, you won't be able to use a matrix formula to pull out the diagonals--you'll need to enter a separate formula for each cell). It is customary to format e.s.d.'s using a custom format such as " \pm General".

15) Correlation coefficients can be calculated in cells H2:J4 by the following matrix formula:

$$= S *(1/ESD_P)*TRANSPOSE(1/ESD_P)$$

Note that regular multiplication is used here. The first multiplication divides each element in the i^{th} row by the esd of P_i . The second multiplication divides each element in the j^{th} column by the esd of P_j . You should verify that the results are in accord with the formulae given in the least-squares handout.

16) By now you have least squares estimates for P_1 , P_2 and P_3 , but what we really want are derived parameters V_{max} , K_m and K_1^c . Solve the simultaneous equations which occur in the definitions of P for equation 1b, and use the equations you get to enter formulae in cells C7:C9 for V_{max} , K_m and K_1^c . Call this 3×1 matrix \mathbf{Q} .

17) The next goal is to calculate a covariance matrix for \mathbf{Q} using \mathbf{S} and the derivatives of \mathbf{Q} with respect to \mathbf{P} . First the array of partial derivatives $\frac{\partial Q_i}{\partial P_j}$ must be calculated. This array is called $\frac{d\mathbf{Q}}{d\mathbf{P}}$. It can be evaluated as follows (I leave it to you to fill in the last row):

$$\frac{d\mathbf{Q}}{d\mathbf{P}} = \begin{pmatrix} \partial Q_1/\partial P_1 & \partial Q_1/\partial P_2 & \partial Q_1/\partial P_3 \\ \partial Q_2/\partial P_1 & \partial Q_2/\partial P_2 & \partial Q_2/\partial P_3 \\ \partial Q_3/\partial P_1 & \partial Q_3/\partial P_2 & \partial Q_3/\partial P_3 \end{pmatrix} = \begin{pmatrix} -1/P_1^2 & 0 & 0 \\ -P_2/P_1^2 & 1/P_1 & 0 \\ \underline{\hspace{1.5cm}} & \underline{\hspace{1.5cm}} & \underline{\hspace{1.5cm}} \end{pmatrix}$$

Fill in the appropriate formulae in cells K7:M9, which can then be named the $d\mathbf{Q}_d\mathbf{P}$ matrix.

18) In cells E7:G9, the covariance of \mathbf{Q} (**covarQ**) can be calculated from the following matrix equation involving successive matrix multiplications (derivation not shown; recall that \mathbf{S} is the covariance of \mathbf{P}).

$$\text{covarQ} = \frac{d\mathbf{Q}}{d\mathbf{P}} \mathbf{S} \left(\frac{d\mathbf{Q}}{d\mathbf{P}} \right)'$$

19) In cells D7:D9, calculate the e.s.d.'s of the \mathbf{Q} parameters, and in cells H7:K9, calculate the correlation coefficients. These calculations will be very similar to those in steps 14 and 15.

20) Write down (or print out) the results you have obtained on the unweighted least-squares fit. Save your spreadsheet with a name something like "UnwtdCompetitive".

Before going on, stop to think a bit about the results from this unweighted linear least squares refinement, and what might be some problems with this approach. If you know how to make graphs in EXCEL, it may help to make a plot of $1/V$ vs $1/[S]$ (see Stryer for what this graph should look like), perhaps including both $\mathbf{y} = 1/V_{\text{obs}}$ and \mathbf{y}_{calc} on the same graph. (The \mathbf{y} can be graphed using symbols for the data points, but no lines connecting them. The \mathbf{y}_{calc} is customarily plotted as a line, but in this case it is probably best to plot the data in sections corresponding to different values of $[I]$ so that there will be a different line for each $[I]$. This is tricky. If you were able to download the spreadsheet template containing the data (<http://www.haverford.edu/chem/scarro/leastsquares/enzymeA.xls>), a graph of this type was set up for you.

Exercise 2: Weighted least squares analysis of Dataset A, assuming competitive inhibition.

Perhaps you replicated one particular experiment 5 times for each [I] and [S]), and found using your calculator that the e.s.d. of the values was 0.1 nmol/minute. For this exercise, we will assume that the random error associated with each V measurement (which might be based on a rate of spectrophotometric absorbance change) was the same, regardless of [I] or [S].

2.1) Copy your Exercise 1 spreadsheet and give it a name like “WtedCompetitive”. Open it with EXCEL and modify it (as described below) in order to do weighted least squares.

2.2) Enter the estimated errors (0.1) in cells D12:D41, just after the V_{obs} values.

2.3) Even though the esd of all V measurements are considered to be the same, there will be different errors associated with $1/V = y$ values. Differential calculus can be used to relate the e.s.d.’s of two quantities which are functionally related:

$$\text{esd}(y) = \left| \frac{dy}{dV} \text{esd}(V) \right| = \frac{1}{V^2} \text{esd}(V)$$

Use cells G12:G41 of your spreadsheet to calculate esd(y) from this formula.

2.4) In cells K12:K41, calculate a vector (30×1 matrix) of the diagonal elements of the **W** matrix. Recall that these are $1/\sigma_y^2$. For instance, $K12 = 1/G12^2$. Call this vector **diagW**.

You might wonder why we don’t write out the entire **W** matrix (30×30), with all the zeros for off diagonal elements. This would be a lot of work, and would take up lots of space on our spreadsheet. However, in EXCEL, the matrix formula $\{=\text{diagW} * X\}$ will yield the result **W X**. The way EXCEL treats ordinary multiplication (*) of matrices is not standard mathematical notational practice, but we can make use of it to ease our calculations.

2.5) Change the matrix equations for Rdata and Resid in cells B1 and B2. The weighted Resid = $\Sigma[(y_{\text{diff}})/\sigma]^2$, and a similar modification applies for Rdata. For your edification, write down the value of R_{weighted} and GOF using the old estimates of parameters from unweighted least squares. This number should later drop as the weighted least squares estimates are calculated.

2.6) In order to change to a weighted least squares estimation of P, you must change the expressions for **M** and **U** to include the effect of multiplication by the **W** matrix. For instance, the new expression for **M** is $\{=\text{MMULT}(\text{TRANPOSE}(X), \text{diagW} * X)\}$. Note that you must select the entire matrix (cells K2:M4 for **M**) in order to change its formula.

2.7) I’ll let you work out the new expression for **U** from your notes.

2.8) Formulae for calculating the e.s.d.’s of V_{max} , K_m and K_i should stay the same, so the weighted least squares results should now be displayed on the spreadsheet. Write down your results, and note how they have changed. The Resid and GOF values should have decreased from when you wrote them down in step 25, indicating a better fit than obtained with non-weighted least squares refinement. If you made a graph before, you can print it out again for comparison to that from Exercise 1.

2.9) Save your work. If you forgot to save the unweighted spreadsheet, you can easily get back to your unweighted least squares fit by replacing the formulae in the **esdY** column with a series of ones (1). “Unweighted” can also be interpreted that everything gets the same weight of unity.

Exercise 3: Weighted linear least squares assuming uncompetitive inhibition

3.1) Make a copy of the weighted competitive spreadsheet; call the copy “WtedUncompetitive” and make the following modifications:

3.2) Look back at the boxed equation 3b (uncompetitive) in the introduction to this handout. Note that only x_3 and P_3 have changed relative to equation 1b (competitive). This means that only a few places in the spreadsheet need to be changed:

- The third column of the **X** matrix needs to be changed so that it displays [I] rather than [I]/[S].
- The third rows (corresponding to K_i) of the **Q** vector and **dQ_dP** matrix need to be changed. Go back and repeat steps 16 and 17 from exercise 1 to figure out the new formulas.

That’s all that needs to be done, and you should have new least-squares estimates of parameters shown on the spreadsheet. If you have a graph of the results, you will probably notice that the fits are less convincing than they were for competitive inhibition. The GOF and R values are also much higher now, indicating a worse fit.

The take-home lesson here is that just because parameters are estimated from a least-squares refinement, it does not mean that those values are applicable; the wrong theoretical model may have been used. A good indicator of an improper model is when the GOF is much larger than 1; values larger than 2 should raise warning flags. (Beware of relying solely on the GOF, because sometimes experimenters conclude when they get poor fits that their data really wasn’t as good as they had thought, so they increase the values of σ_{data} , and this will have the effect of decreasing the GOF).

Exercise 4: Weighted linear least squares assuming mixed inhibition.

Unfortunately, it isn't as easy to modify the spreadsheet to handle mixed inhibition, because the sizes of the matrices (\mathbf{X} , \mathbf{M} and θ , for instance) will be a bigger (4 parameters, not 3). EXCEL will not allow you to insert rows or columns in the middle of a matrix formula. You could test your understanding of the least squares method by starting from scratch with a new spreadsheet.

Or, you can save time by making use of an EXCEL macro sheet available from <http://www.haverford.edu/chem/scarrow/leastsquares/lsmacro.xls>. Open this with EXCEL, choose **Macro>Macros** from the **Tools** menu, and run the "construct_ws" macro (in earlier versions of EXCEL (vs. 4 for instance), choose **Run** from the **Macros** menu). You will be ...

- asked if you want to do non-linear least squares (answer no by clicking "Cancel")
- asked if you want to calculate dependent parameters (answer yes by clicking "OK"; this creates the \mathbf{Q} and $\mathbf{dQ}_{\mathbf{dP}}$ matrices).
- prompted for n_{data} (= 30 for our example)
- prompted for n_{p} (= 4 for the mixed inhibition case)
- prompted for n_{scratch} (= 5 columns needed for $[\mathbf{I}]$, $[\mathbf{S}]$, V_{obs} , $\text{esd}(V)$ and V_{calc}). The "scratch" columns are placed to the left of the \mathbf{Y} and \mathbf{X} matrices, and include information that is used to calculate these matrices (and anything else you want to put there).

You will need to copy the data into these 5 "scratch" columns from the earlier spreadsheets. Also copy \mathbf{y} , $\text{esd}\mathbf{y}$, and \mathbf{X} (first three columns) from the earlier competitive spreadsheet. (Note that, since these columns are contiguous you can do all of this copying at once.) The fourth column of \mathbf{X} is, of course, new, and a new formula will need to be entered in this column. You will also need to enter the new formulae into the \mathbf{Q} vector and $\mathbf{dQ}_{\mathbf{dP}}$ matrix.

Record the values and uncertainties of the four dependent parameters of interest (the \mathbf{Q} and $\text{esd}_{\mathbf{Q}}$ vectors). If you have entered all of the formulae correctly, you will find that the uncertainty in K_{I}^{M} is larger than its value. This argues against the mixed mode of inhibition.

Note that the residual R is decreased slightly from that calculated with competitive inhibition. Does this indicate that the mixed inhibition model is justified? The statistical answer is that lower R does not automatically mean a more likely fit if the R was lowered by inclusion of additional refined parameters. Adding an additional parameter to the model (in this case adding in K_{I}^{M}) almost always will lower R at least slightly. The "goodness of fit" GOF parameter is defined in such a way to compensate for this expected decrease; the denominator is $n_{\text{data}} - n_{\text{p}}$, so that if the R value did stay the same when an additional parameter was added to the model (n_{p} increased by 1), then the GOF would increase to indicate a less favored fit.

Comparing the GOF of the mixed inhibition model to that of the competitive inhibition model offers an objective way of deciding which model is more likely (that with the lower GOF is more likely). From your results, which of these models is more likely? Of course, if the two values of GOF are close (say within 20% of each other) it is probably not reasonable to rule out the possibility of either model based on the fits.

Because the refined K_{I}^{M} is much larger than the refined K_{I}^{C} , can you rule out non-competitive inhibition (where the two inhibition constants are equal)?

Exercise 5 : Non-linear least squares estimation of parameters using equations (1) - (4) directly.

For this exercise, I give you a new set of data (Dataset B, below). The task is to decide whether the inhibition is competitive, non-competitive, uncompetitive or mixed based on which equation (1 - 4) gives the lowest GOF to the data. Because equation 2 cannot be linearized, we will use non-linear least squares techniques to obtain the parameters.

Dataset B: (download at <http://www.haverford.edu/chem/scarrow/leastsquares/enzymeB.xls>)

$V_{\text{obs}} =$	no inhibitor	[I] = 0.3 mM	[I] = 1.0 mM	[I] = 3.0 mM	[I] = 10.0 mM
[S] = 0.1 mM	0.8 nmol/min	0.7	0.7	0.5	0.1
[S] = 0.2 mM	1.5	1.5	1.1	0.8	0.3
[S] = 0.3 mM	2.1	2.0	1.6	1.0	0.3
[S] = 0.4 mM	2.7	2.4	1.8	1.2	0.5
[S] = 0.6 mM	3.6	3.1	2.6	1.6	0.6
[S] = 1.0 mM	4.7	4.0	3.3	2.0	1.0

Start, as before, assuming competitive inhibition (equation 1). You need to look at the data and come up with reasonable guesses for V_{max} , K_m , and K_i . Good guesses can be obtained by remembering that K_m is the [S] at which $V = V_{\text{max}}/2$ (no inhibitor) and the K_i 's will be the same order of magnitude as the IC_{50} 's (concentrations where the rates are slowed down by 50%).

Use the macro (<http://www.haverford.edu/chem/scarrow/leastsquares/lsmacro.xls>) to set up a non-linear least-squares worksheet with $n_{\text{data}} = 30$, $n_{\text{p}} = 3$ and $n_{\text{scratch}} = 2$. There is no need to calculate dependent parameters this time. The spreadsheet looks pretty much the same, except there are two extra columns in both the top and main regions of the spreadsheet.

5.1) Put the [I] and [S] values in the two "scratch" columns. The V_{obs} values are now the y_{obs} matrix. Because we are doing non-linear least squares, there is no need to calculate $1/V_{\text{obs}}$ (unless you want to make a graph of $1/V$ vs $1/[S]$, in which case you may use the empty columns at the right of the spreadsheet).

5.2) Enter the esd of the y values in column F: as before, assume that each measurement has the same uncertainty of ± 0.1 (nmol/min).

5.3) Enter the guessed parameters (V_{max} , K_m , and K_i) in the 3×1 vector C1:C3 called $\mathbf{P}_{\text{guess}}$.

5.4) The values of V_{max} , K_m , and K_i in this vector are used to calculate y_{g} --you must fill in the formulae in the y_{g} column to do this. Base the formula on equation 1 (for competitive inhibition), using the absolute references $\$C\2 , $\$C\3 and $\$C\4 to refer to V_{max} , K_m , and K_i . Be careful to put parentheses in the right place: it is worthwhile to check the formula by hand-calculating a few cells.

5.5) The \mathbf{X} matrix must be filled in. Note that \mathbf{X} refers to the derivative of y_{g} as the various $\mathbf{P}_{\text{guess}}$ are changed. You need to use your calculus skills to derive the derivatives (dV/dV_{max} , dV/dK_m , dV/dK_i), and then implement the formulae in your spreadsheet. *Caution--derivatives are tricky and this is usually where I make errors. I usually check my derivatives by making small changes in $\mathbf{P}_{\text{guess}}$ and recording what happens to y_{guess} and then verifying that*

$\Delta y_{\text{guess}}/\Delta \mathbf{P}_{\text{guess}} \approx dy/dP$. In checking derivatives, I always choose a row corresponding to a data point which isn't a "special case"--for instance, I don't use a row with $[I]=0$.

This completes the setup of the non-linear least squares spreadsheet.

The \mathbf{y} that is used for the non-linear least squares calculation is actually a $\Delta\mathbf{y}$ --the difference between \mathbf{y}_g and \mathbf{y}_{obs} . What the spreadsheet calculates now is the estimated correction that will be needed to the \mathbf{P}_{guess} values in order that \mathbf{y}_g be made closer to \mathbf{y}_{obs} (on average). Thus, a new guess, \mathbf{P}_{new} , can be calculated as $\mathbf{P}_{new} = \mathbf{P}_{guess} + \Delta\mathbf{P}$, where $\Delta\mathbf{P}$ is calculated from exactly the same formulae as in the linear least squares case (browse the spreadsheet to convince yourself of this).

After you have filled in all of the information (steps 5.1 – 5.5), the $\Delta\mathbf{P}$ will be calculated, as will \mathbf{P}_{new} , the new “guess” as to the true parameters. Non-linear least squares is an iterative process. In your notebook, record values of \mathbf{P}_{guess} , \mathbf{P}_{new} , Resid and Rdata. Then change the values in \mathbf{P}_{guess} to those calculated in \mathbf{P}_{new} . A convenient way of doing this is using “Copy” (when \mathbf{P}_{new} is selected) and “Paste Special (values)” (when \mathbf{P}_{guess} is selected). After the first cycle, you can use “Repeat Paste Special” (from the Edit menu – also available as the Redo icon or as cmd-Y (Mac) or cntrl-Y (Windows)), to run successive cycles of least squares.

Unless $\Delta\mathbf{P}$ is very small, the “Resid” which is calculated will be inaccurate (it is usually *underestimated*). Thus, it is not uncommon for “Resid” to actually increase during least squares. For non-linear least squares, the “Rdata” is the residual which is minimized by the process of iterative cycling. After several cycles, $\Delta\mathbf{P}$ will be very small, and Rdata = Resid. At this point the least-squares refinement is said to have *converged*. You should always ensure the refinement has converged before reporting values.

The results of the non-linear least squares include the best estimates of \mathbf{P} values ($\mathbf{P}_{new} = \mathbf{P}_{guess}$ at convergence) and their uncertainties (\mathbf{esd}_P), as well as correlation coefficients. These should be reported along with the residual and GOF (correlation coefficients only need to be reported if they are close to ± 1 – in this case, report any coefficients outside of the range ± 0.9).

In some cases, the refinement may not converge. The refinement isn't converging if Rdata (not Resid) is increasing from one cycle to the next. In other cases, a parameter such as a K_i or K_m may refine to a meaningless negative value. If this happens, try repeating the refinement procedure using different initial guess of the parameters. If you want to see what not-converging looks like, try using \mathbf{P}_{guess} which are off by several orders of magnitudes. *Failure to converge may also indicate an error in the derivative formulae entered for the X matrix.*

It is theoretically possible in some least squares problems that if the initial guesses are way off, one finds a "local minimum" which is a convergence of the least squares procedure, but which has a significantly higher R than found for the "global minimum" solution that you'd like to find. This is a known problem with non-linear least squares.

Set up four different spreadsheets assuming competitive, non-competitive, uncompetitive and mixed inhibition and use the GOF criterion to decide which model is most likely. Report the refined parameters from this model along with e.s.d.'s of the parameters.