Global nonlinear regression with Prism 4

Harvey J. Motulsky (GraphPad Software) and Arthur Christopoulos (Univ. Melbourne)

Introduction

Nonlinear regression is usually used to analyze data sets one at a time. Global nonlinear regression fits an entire family of data sets at once, sharing one or more parameters between data sets. For each shared parameter, global nonlinear regression finds one (global) best-fit value that applies to all the data sets.

Global curve fitting has been available for decades, but is used infrequently by biological researchers. To give you a sense of how versatile this technique is, we present here five examples of pharmacology experiments that are usually analyzed by fitting curves one at a time, but could also be analyzed by global curve fitting.

How global nonlinear regression works

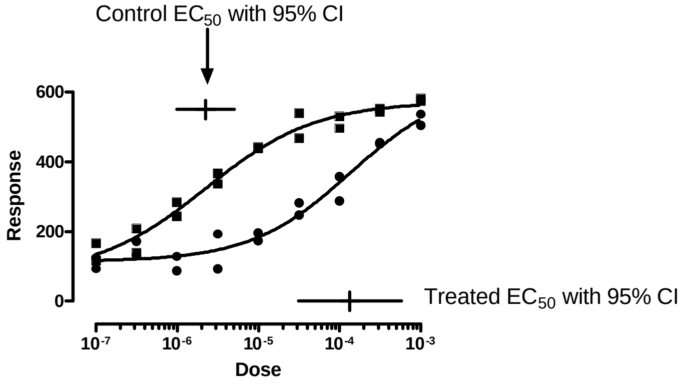
Nonlinear regression finds parameters of a model that make the curve come as close as possible to the data. This is done by minimizing the sum of the squares of the vertical distances between the data points and curve. Mathematical statisticians have proven that if the scatter of data points around the curve follows a Gaussian distribution, the parameter values that minimize the sum-of-squares are those that are most likely to be correct.

Global nonlinear regression extends this idea to fitting several data sets at once. Calculate the sum-of-squares of the curves from each data set, and then add them to compute a total. Global nonlinear regression minimizes this total sum-of-squares.

Global fitting only makes sense when all the data are expressed in the same units. If different data sets are expressed in different units, be cautious about using global fitting. The problem is that your decision about which units to use can change the results. For example, imagine what happens if you change one data set from expressing weight in grams to expressing weight in milligrams. All the values are now increased by a factor of one thousand, and the sum-of-squares for that data set is increased by a factor of one million (one thousand squared). Compared to other data sets, expressed in different units, this data set now has a much greater impact on the fit. If you really need to do global fit to data sets using different units, consider first normalizing the data so they are comparable.

Example 1. Fitting incomplete data sets.

The graph below shows two dose-response curves. The goal of the experiment is to determine the two EC_{50} values. The EC_{50} is the concentration (dose) that gives a response half-way between the minimum and maximum responses. Each curve in the graph below was fit individually to one of the data sets. The horizontal lines show the 95% confidence interval of the EC_{50}.



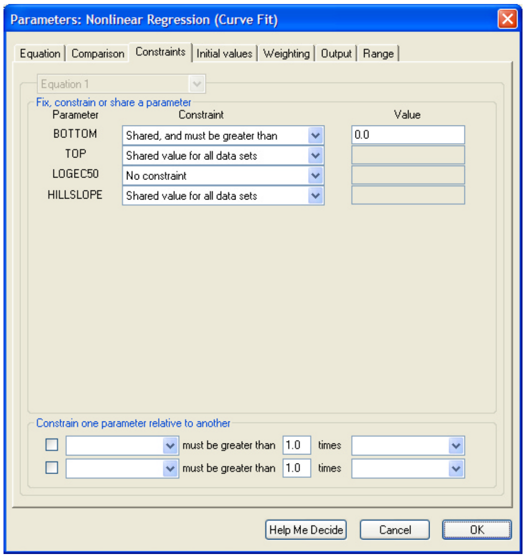
While the curves nicely fit the data points, the confidence intervals are quite wide. We really haven't determined the EC_{50} with sufficient precision to make useful conclusions. The problem is that the control data don't really define the bottom plateau of the curve, and the treated data don't really define the top plateau of the curve. Since the data don’t define the minimum and maximum responses very well, the data also don't define very clearly the point half-way between the minimum and maximum responses. Accordingly, the confidence intervals for each EC_{50} extend over more than an order of magnitude.

The whole point of the experiment was to determine the two EC_{50} values, and (with this analysis) the results were disappointing. There is an unacceptable amount of uncertainty in the value of the best-fit values of the EC_{50}.

One way to determine the EC_{50} values with less uncertainty is to redo the experiment, collecting data over a wider range of doses. But it is also possible to get better results from the data we have. One problem is that we don't know the bottom of the curve. If we had control data with zero dose, we could fix the bottom plateau to those control values. Another approach would be to average the first few responses in the control (left) curve and define this average to be the minimum response. Then average the last few responses in the treated (right) curve and define this average to be the maximum response. Then fit the two curves separately, fixing the top and bottom of the curve to constant values defined by the minimum and maximum responses. You'll get reasonable results this way, but it is a bit arbitrary. Do you define the minimum based on just the first concentration, the mean of the first two, the mean of the first three… ? You'll encounter the same problem if you normalize the responses from 0 to 100, and then fix the bottom plateau to equal 0 and the top plateau to equal 100. Before you can do this normalization, you must decide which values define the top and bottom of the curve.

You can get much better results from the original set of data by analyzing the data more sensibly, using global curve fitting. When you use global curve fitting, you have to tell the program which parameters to share between data sets and which to fit individually. For this example, we’ll instruct the program to find one shared best-fit value of the top plateau that applies to both data sets, one shared best-fit value of the bottom plateau that applies to both data sets, and one shared best-fit value of the slope factor (how steep is the curve) that applies to both data sets. Of course, we won’t ask the program to share the EC_{50} value. We want the program to determine the EC_{50} separately for control and treated data.

To do global fitting with Prism 4, you specify which parameters are to be shared on the constraints tab. In this example, we'll also constrain the parameter Bottom to be greater than zero.



Fix, constrain or share a parameter

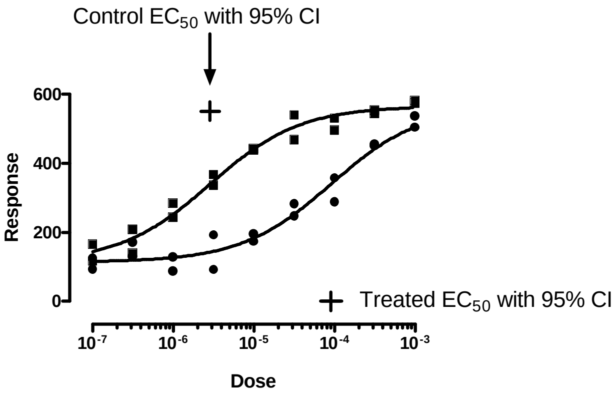
Parameter	Constraint	Value
BOTTOM	Shared, and must be greater than	0.0
TOP	Shared value for all data sets	
LOGEC50	No constraint	
HILLSLOPE	Shared value for all data sets	

Constrain one parameter relative to another

□ must be greater than 1.0 times

□ must be greater than 1.0 times

Here are the results.

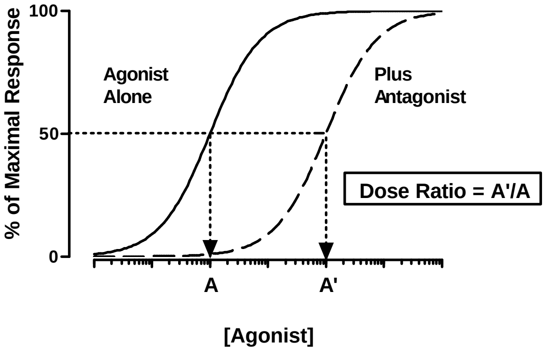


The graph of the curves looks only slightly different. But now the program finds the best-fit parameters with great confidence. Each EC_{50} value is determined, with 95% confidence, within a factor of two (compared to a factor of ten or more when the curves were fit individually). We’ve accomplished the goal of the experiment, to determine the two EC_{50} values with reasonable certainty.

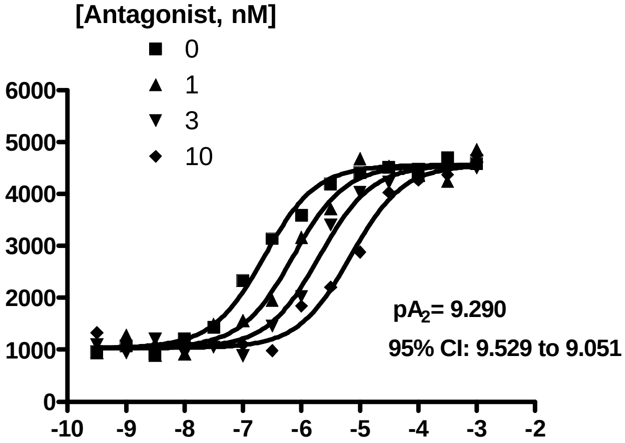
The control data define the top of the curve pretty well, but not the bottom. The treated data define the bottom of the curve pretty well, but not the top. By fitting both data sets at once, using a global model, we are able to determine both EC_{50} values with reasonable certainty.

Example 2. Shift of dose-response curve in the presence of an antagonist.

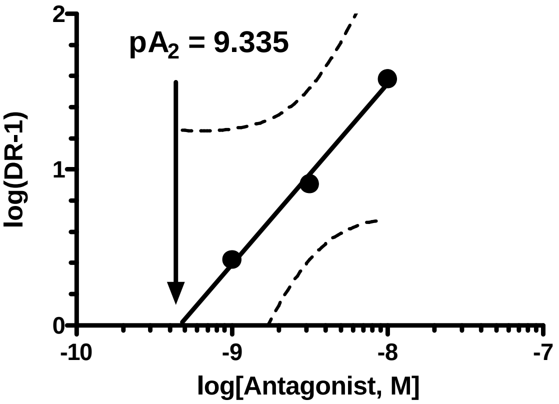
If you perform a dose-response curve in the presence of a competitive antagonist, the dose-response curve is shifted to the right as shown below. You are not only interested in the individual EC_{50} values. You want to know how the EC_{50} changes when you increase antagonist concentration, as this lets you determine the affinity of the antagonist (the pA_2).



The conventional approach is to fit each dose-response curve individually, determining an EC_{50} value for each.



Then, for each concentration of antagonist, compute the dose ratio (DR) which is the EC_{50} of the agonist in the presence of the antagonist divided by the EC_{50} with agonist alone. The relationship between DR and concentrations of antagonists are then plotted in a Schild plot.



If you set DR to 2.0, the log(DR-1) equals zero. Thus the X intercept of the linear regression line is the logarithm of the concentration of antagonist required to shift the agonist dose-response EC_{50} by a factor of two. If you make additional assumptions (competitive binding, Hill slope of 1.0) this value is the equilibrium dissociation constant for antagonist binding. In this example, the concentration of antagonist required to shift the agonist dose-response curve by a factor of two is $10^{-9.335}$M, or 0.463 nM. Multiply the logarithm of that value by -1 to get the pA_2, which is 9.335.

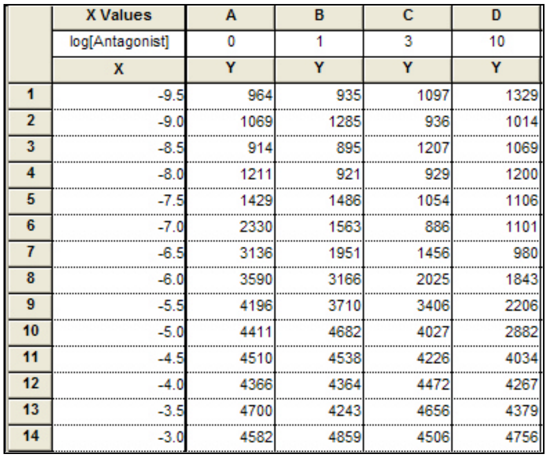
When you analyze data, you don't just want to know the best-fit value of a parameter (in this case the pA_2) but also its 95% confidence interval so you know how precisely you have determined the parameter. The Schild plot in the lower panel above shows the 95% confidence band of the linear regression line as dashed curves. You can see that the upper confidence band never crosses the axis at Y=0. This means that the lower confidence limit for the X intercept is simply not defined. While the Schild analysis gives us a value for the pA_2, it is not able to quantify its precision in this example.

Using global fitting, you can analyze all the data at once, saving the hassles of calculating dose-ratios and creating a Schild plot. The results of global fitting are also more precise, and you get a confidence interval as well as the best-fit value of the pA_2. Here is the equation describing the response of an agonist in the presence of a competitive antagonist.

$$\text{Y = Response = Bottom} + \frac{(\text{Top-Bottom})}{1+\left(\frac{10^{\text{LogEC}_{50}}\left[1+\left(\text{[Antagonist]}/10^{-\text{pA}_2}\right)^{S}\right]}{10^{X}}\right)^{\mathit{HillSlope}}}$$

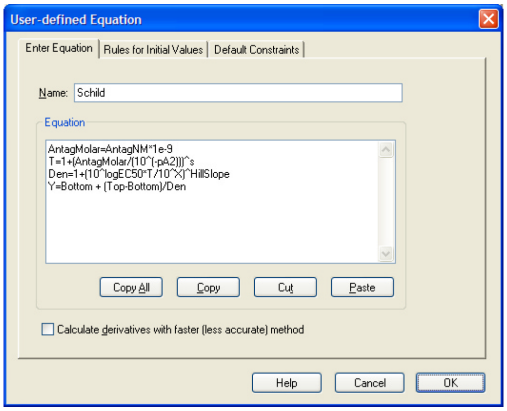
The Y values are the responses. The X values are the log of agonist concentration. The concentration of antagonist is a constant for each curve, but differs between curves. In Prism, you enter these values as column titles, and then define the parameter to be a data set constant. HillSlope is the slope of the agonist-only dose-response curve. S is the Hill Slope of antagonist binding. For this example, we'll fix S to 1.0, assuming competitive one-site binding.

Here is the data table. Note that the column titles are labeled with the antagonist concentrations (in nM for this example). You'll see in a moment how these are read by Prism.



	X Values	A	B	C	D
	log[Antagonist]	0	1	3	10
	X	Y	Y	Y	Y
1	-9.5	964	935	1097	1329
2	-9.0	1069	1285	936	1014
3	-8.5	914	895	1207	1069
4	-8.0	1211	921	929	1200
5	-7.5	1429	1486	1054	1106
6	-7.0	2330	1563	886	1101
7	-6.5	3136	1951	1456	980
8	-6.0	3590	3166	2025	1843
9	-5.5	4196	3710	3406	2206
10	-5.0	4411	4682	4027	2882
11	-4.5	4510	4538	4226	4034
12	-4.0	4366	4364	4472	4267
13	-3.5	4700	4243	4656	4379
14	-3.0	4582	4859	4506	4756

Here is the equation entered as a user-defined equation, followed by the rules for initial values.

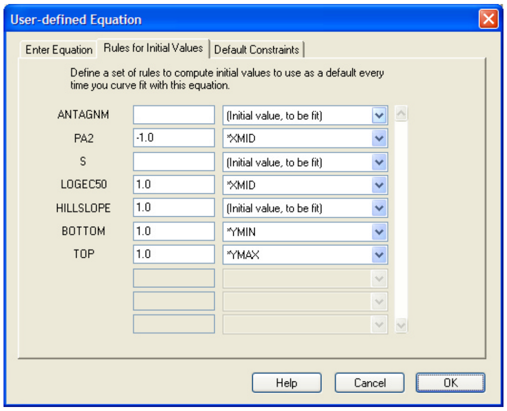


Name: Schild

Equation

```
AntagMolar=AntagNM*1e-9
T=1+(AntagMolar/(10^(-pA2)))^s
Den=1+(10^logEC50*T/10^X)^HillSlope
Y=Bottom + (Top-Bottom)/Den
```

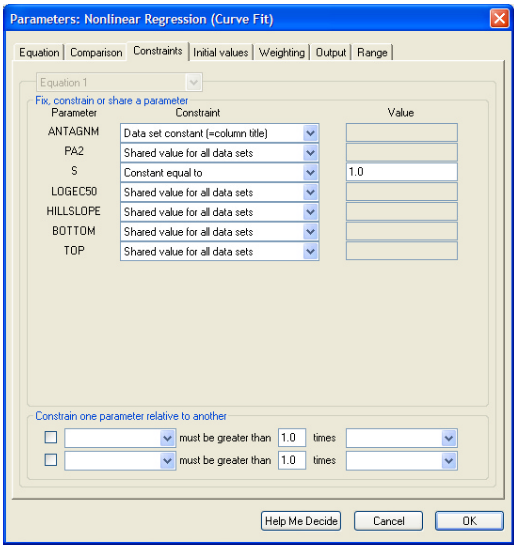
□ Calculate derivatives with faster (less accurate) method



Define a set of rules to compute initial values to use as a default every time you curve fit with this equation.

ANTAGNM		(Initial value, to be fit)
PA2	-1.0	*XMID
S		(Initial value, to be fit)
LOGEC50	1.0	*XMID
HILLSLOPE	1.0	(Initial value, to be fit)
BOTTOM	1.0	*YMIN
TOP	1.0	*YMAX

To use global fitting, we need to specify which parameters are shared. We also need to tell Prism that the antagonist concentrations are in the column titles. This is done on the constraints tab of the nonlinear regression dialog.



Fix, constrain or share a parameter

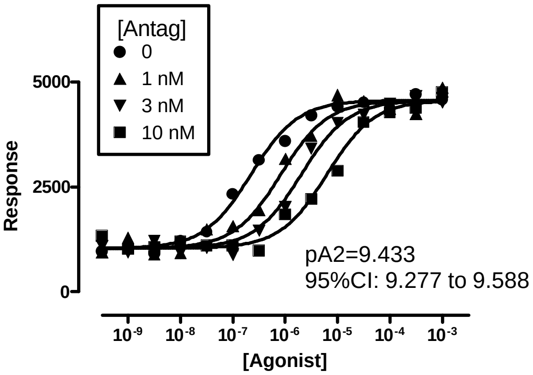
Parameter	Constraint	Value
ANTAGNM	Data set constant (=column title)	
PA2	Shared value for all data sets	
S	Constant equal to	1.0
LOGEC50	Shared value for all data sets	
HILLSLOPE	Shared value for all data sets	
BOTTOM	Shared value for all data sets	
TOP	Shared value for all data sets	

Constrain one parameter relative to another

□ must be greater than 1.0 times
□ must be greater than 1.0 times

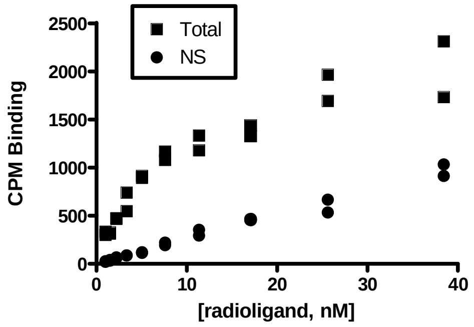
We defined the parameter ANTAGNM (antagonist concentration in nM) to be a data set constant, whose values for each data set comes from the column title of that data set. We set S equal to a constant of 1.0. We are assuming that the antagonist binds competitively. We share all the other parameters, so their values are determined by all the data at once.

In this example, the best-fit value of pA_2 is 9.43, with a 95% confidence interval ranging from 9.28 to 9.59.



Example 3. Total plus nonspecific binding.

In this example, we measured equilibrium binding of radioligand at various concentrations of radioligand to find the B_{max} and K_d of the radioligand. Since the ligand binds to nonspecific sites as well as the receptor of interest, you also measure nonspecific binding (binding of radioligand in the presence of an excess of an unlabeled receptor blocker).



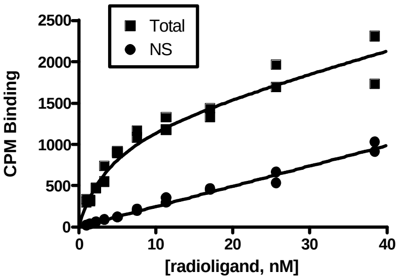
These kind of data are usually analyzed by first subtracting the nonspecific binding from the total binding. The resulting specific binding is then fit to a model that describes equilibrium binding to one receptor site.

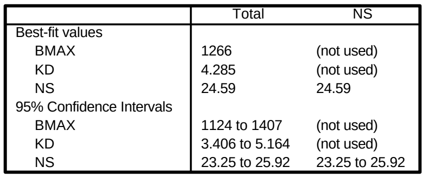
Global fitting can simultaneously fit both the total binding and the nonspecific binding. There is no need to first subtract the two data sets. The only trick is to write a model that fits different equations to each data set. Prism lets you do this, as shown below.

```
Specific=Bmax*X/(Kd+X)
Nonspecific=NS*X
<A>Y=Specific + Nonspecific
<B>Y=Nonspecific
```

The line preceded by <A> only applies to the first data set (column A, total binding). The line preceded by <B> only applies to the second data set (nonspecific binding). If you precede a line with <~A> it applies to all lines except the first.

Here is the result. We used global nonlinear regression to fit both data sets at once, sharing the parameter NS.



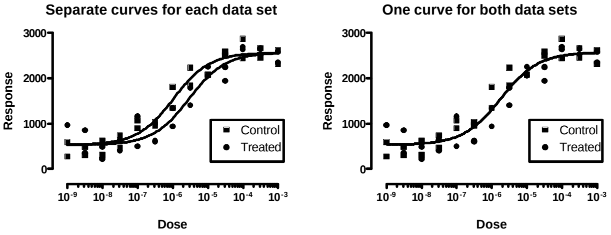


	Total	NS
Best-fit values		
BMAX	1266	(not used)
KD	4.285	(not used)
NS	24.59	24.59
95% Confidence Intervals		
BMAX	1124 to 1407	(not used)
KD	3.406 to 5.164	(not used)
NS	23.25 to 25.92	23.25 to 25.92

Example 4. Comparing data sets

Global fitting can be used to compare curves between data sets. The example below shows a dose-response curve collected under control and treated conditions. We want to know whether the difference between the two dose-response curves is convincing.

The panel on the left assumes that the treatment was effective. It fits the two logEC50 separately, but shares the parameters that define the bottom, top and slope of the curve. The panel on the right assumes that the treatment is ineffective. It shares all four parameters, so finds one curve for all the data. This is equivalent to entering all the data as one big data set. The fit on the left has a smaller sum-of-squares (the points are closer, on average, to the separate curves), but also has an extra parameter (since it fits two EC50s, rather than one).



Prism offers two ways to compare the two fits.

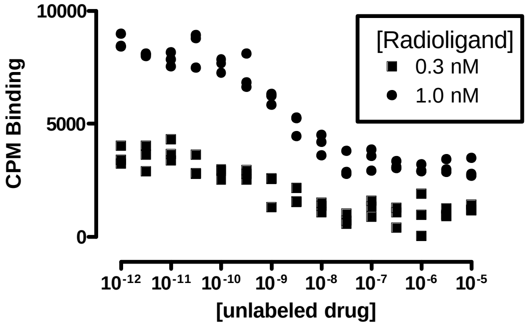
The *extra sum-of-squares F test* starts with an assumption, also known as the null hypothesis. This assumption is that the treatment is really ineffective. The test then asks: If this assumption were true, what is the chance that the model on the left would fit your data as much better as it does? In other words, what is the chance that you'd see such a large difference in sum-of-squares? The answer, the P value, is 0.0024. Since this is so small, we conclude that the assumption is likely to be wrong. Instead we conclude that the treatment had a statistically significant effect on the EC50.

Prism can also compare the two fits using *Akaike's Information Criterion* (AIC). This procedure answers the question: Which model (right or left panel) is more likely to be correct, and how much more likely? The answer is that the model on the left is 47 times more likely to be right. There is a 2.1% chance that the model on the right is correct, and a 97.9% chance that the model on the left is right.

Example 5. Homologous binding

You want to know how many of a particular kind of receptor your tissue sample has, and how tightly a particular drug (ligand) binds to those receptors. Since the radioligand is quite expensive, you don't vary its concentration. Instead, you add a single concentration of a radioactively labeled drug to all the tubes, and also add various amounts of the same drug that is not radioactively labeled. You assume that the two forms of the ligand bind identically to the receptors, that you have reached equilibrium, that the ligand only binds to one kind of receptor, and that there is no cooperativity. As you add more of the unlabeled ligand, more of it binds to the receptors so less of the radioactive ligand binds. Since you only measure binding of the labeled (radioactive) ligand, you see a downhill binding curve. You want your analysis to find the affinity of the receptors for the drug (K_d) and also the maximum number of binding sites (B_{max}).

Here are some sample data. This experiment was run with two different concentrations of the radioligand. This is unusual (most homologous binding experiments use a single concentration of radioligand), and you'll see the advantage of using two concentrations later in this example.



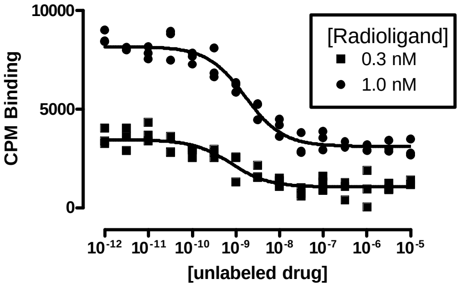
From the law of mass-action, it is easy to derive an equation that describes the binding.

$$Y=\frac{B_{max}\times[Hot]}{[Hot]+[Cold]+K_d}+NS$$

The concentration of hot ligand (the parameter [Hot]) is fixed by your experimental design, and is the same for all tubes. The concentration of cold (the parameter [Cold]) is also set by the experimenter, and varies from tube to tube. In order to find more appropriate confidence intervals of the K_d, it is better to actually fit the logarithm of the K_d (LogKd), using this equation.

$$Y=\frac{B_{max}\times[Hot]}{[Hot]+[Cold]+10^{\log K_d}}+NS$$

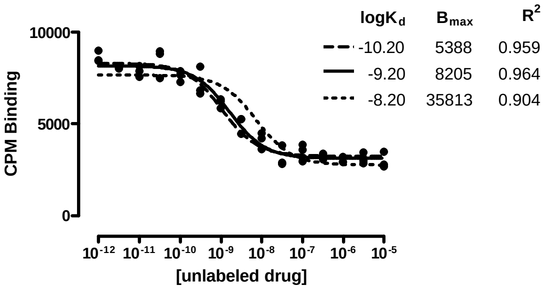
Here are the best-fit curves, each determined independently.



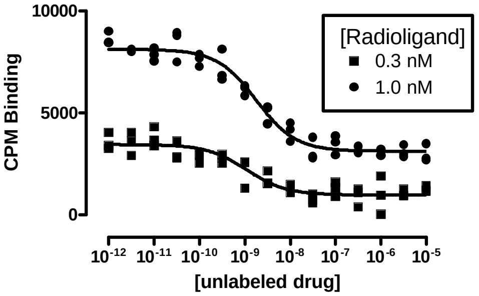
	0.3	1.0
Best-fit values		
LOGKD	-9.331	-9.204
BMAX	6105	8205
HOTNM	0.3000	1.000
NS	1062	3121
95% Confidence Intervals		
LOGKD	-9.834 to -8.829	-9.579 to -8.829
BMAX	1851 to 10360	5469 to 10941
NS	861.5 to 1263	2909 to 3333

Note that the B_{max} is the total possible specific binding (in cpm) to all binding sites. At the concentration of radioligands used in this experiment, the radioligand binds to only a fraction of these sites. That explains why the best-fit B_{max} values (6125-8205 cpm) are quite a bit higher than the observed specific binding (about 1000-4000 cpm).

Note that the confidence intervals are quite wide. The confidence interval for K_d extends over an order of magnitude, and the confidence interval for B_{max} extends more than twofold. Why? Because you get almost the same results when you have lots of receptors that bind the drug weakly, or a few receptors that bind the drug tightly. The graph below superimposes three curves through the data collected using 1 nM of radioligand. The solid curve is the same one shown above, with a $\log K_d$ of -9.2 and a B_{max} of 8205. The dashed curve was fit constraining the K_d to be an order of magnitude lower (so $\log K_d$ was fixed to equal -10.2). With this constraint, the best-fit value of B_{max} decreased to 5388. This curve is almost indistinguishable from the best-fit solid curve. The dotted curve constrains the K_d to be an order of magnitude higher than the best fit value (the $\log K_d$ was fixed to equal -8.2). Since the ligand has much lower affinity for the receptors, it must be binding to a small fraction of many more receptors. This explains why the best fit value of B_{max} increases four fold to 25813. The dotted curve clearly fits the data less well than the solid curve, but the difference is not enormous (the R^2 decreases from 0.964 only down to 0.904). What does it mean that these three curves, which look very similar to one another, span two orders of magnitude in receptor affinity? This tells us that homologous binding data are simply ambiguous when you use a single concentration of radiolabeled ligand.



The graph below shows the two curves fit globally. There is only one K_d and one B_{max} no matter how much radioligand you use, so these parameters were shared between the data sets. Only the best-fit value of the nonspecific binding was fit individually for each data set.



	0.3	1.0
Best-fit values		
LOGKD	-9.107	-9.107
BMAX	8899	8899
HOTNM	0.3000	1.000
NS	980.6	3119
95% Confidence Intervals		
LOGKD	-9.271 to -8.943	-9.271 to -8.943
BMAX	7196 to 10602	7196 to 10602
NS	871.7 to 1089	2977 to 3262

Note that the confidence intervals are narrower than they were. The confidence interval for Kd now spans a factor of two, rather than a factor of ten with individual fits. Fitting both curves at once gave us much more reliable (and more accurate) results. Fitting the data globally gave us useful results. Fitting the two curves individually did not.