

CONTENTS

ビジュアルヘルプ - カーブフィッティング (1)	3
概要	3
カーブフィッティングの用語	4
カーブフィッティングの概要	4
反復フィッティング	5
初期推定	5
終了基準	6
カーブフィッティングでのエラー	6
カーブフィッティングのためのデータ	6
Quick Fit メニューを使ったカーブフィッティング	7
Quick Fit メニューの制限	8
Curve Fitting ダイアログを使ったカーブフィッティング	9
シンプルなケース：組み込み関数を使ってフィッティング (Line Fit)	9
関数とデータの選択	11
2つの便利な追加機能：係数の保持と残差の生成	12
自動推定が動作しない場合	14
定数でフィッティング	16
ユーザー定義関数でフィッティング	17
関数の作成	17
ユーザー定義関数の Coefficients タブ	20
ユーザー定義関数が常に使えるようにする	21
ユーザー定義関数を削除する	22
ユーザー定義フィッティング関数の詳細	23
外部関数 (XFUNC) を使ってフィッティング	23
係数ウェーブ	24
誤差	25
宛先ウェーブ	25
宛先なし	26
自動トレース	26
明示的な宛先	27
新しいウェーブ	27

データのサブセットにフィッティング	27
マスクウェーブを使う	29
重み付け	30
比例加重	31
多変量関数へのフィッティング	32
多変量関数に対するデータの部分範囲のフィッティング	34
多変量フィッティングのモデルの結果	35
多変量フィッティングの例	36
Curve Fitting ダイアログの問題	39

ビジュアルヘルプ – カーブフィッティング（1）

概要

Igor のカーブフィッティング機能は、最も優れた分析機能の 1 つです。
そのハイライトをいくつか紹介します。

- 線形および一般的な非線形カーブフィッティング
- 最小二乗法または変数誤差モデルにおける最小直交距離によるフィッティング
- 暗黙的なモデルへのフィッティング
- 一般的なフィッティング用の組み込み関数
- 組み込み関数に対する自動初期値推定
- ユーザー定義の複雑な関数へのフィッティング
- グリッドデータまたは複数列データの任意の独立変数の関数へのフィッティング
- 複数のフィッティング関数へのフィッティング
- ウェーブフォームまたは XY ペアの一部のフィッティング
- エラーの推定値を生成
- 重み付けのサポート

カーブフィッティングの考え方は、データに適合する数学モデルを見つけることです。

特定の形式の関数を選択するときには、理論的な理由があると思います。

カーブフィッティングは、その関数がデータにできるだけ一致するように、特定の係数を求めます。

何千もの関数のうち、データセットに適合するものをカーブフィッティングで探すことはできません。

また、カーブフィッティングを単にデータに滑らかな曲線を表示する目的で使うこともあります。

これは時々うまくいきますが、平滑化や補間も検討すべきで、これについてはヘルプ Smoothing で説明しています。

Igor では 3 種類の関数にフィッティングすることができます：

- ビルトイン関数
- ユーザー定義関数
- 外部関数 (XFUNC)

組み込みのフィッティング関数は、線形、多項式、正弦、指数、二重指数、ガウス、ローレンツ、ヒル方程式、シグモイド、ロジスティック、対数、Gauss2D（2 次元ガウスピーク）、Poly2D（2 次元多項式）です。

これらの関数に関する詳細は、「組み込みのカーブフィッティング関数」のセクションを参照してください。

ユーザー定義関数は、New Fit Function ダイアログにその関数を入力することで作成します。

非常に複雑な関数は、Procedure ウィンドウに入力する必要があるかもしれません。

外部関数である XFUNC は、C または C++ で記述されます。

XFUNC を作成するには、オプションの Igor XOP Toolkit と C/C++ コンパイラが必要です。

WaveMetrics または他のユーザーから入手した XFUNC を使う時には、ツールキットは必要ありません。

カーブフィッティングは次のような形式の方程式で機能します： $y = f(x_1, x_2, \dots, x_n)$ 。

独立変数 (x_n) の数がいくつであっても関数をフィッティングさせることは可能ですが、ほとんどのケースは 1 つだけです。

多変量フィッティングの詳細については、「多変量関数へのフィッティング」のセクションを参照してください。

また、暗黙的関数へのフィッティングも可能です。暗黙的関数は次の形式です： $f(x_1, x_2, \dots, x_n)$ 。

詳細は、「暗黙的関数へのフィッティング」のセクションを参照してください。

線形制約付きのカーブフィッティングを行うこともできます（「制約付きフィッティング」のセクションを参照してください）。

カーブフィッティングの用語

組み込みのフィッティングは、CurveFit コマンドによって実行されます。

ユーザー定義のフィッティングは、FuncFit または FuncFitMD コマンドによって実行されます。

「カーブフィッティングコマンド」という言葉は、CurveFit、FuncFit、FuncFitMD のいずれか適切なものを指します。

外部関数へのフィッティングは、ユーザー定義関数へのフィッティングと同じように動作します（ただし、Curve Fitting ダイアログに関するいくつかの注意点があります）。

「外部関数（XFUNC）へのフィッティング」を参照してください。

Curve Fitting ダイアログを使う場合は、組み込み関数とユーザー定義関数の区別について詳しく知る必要はありません。

外部関数と、その他のタイプの区別については、少し知っておく必要があるかもしれません。

これは以降で改めて説明します。

カーブに当てはめる数値を「係数」と呼びます。

コマンドや関数に渡す値には「パラメーター」という用語を使います。

カーブフィッティングの概要

カーブフィッティングでは、未知の係数を持つ関数と生データがあります。

係数の値を求め、その関数が生データにできるだけ一致するようにしたいわけです。

係数の「最良の」値は、カイ二乗値を最小化する値です。

カイ二乗値は次の式で定義されます：

$$\sum_i \left(\frac{y - y_i}{\sigma_i} \right)^2$$

ここで、 y は与えられたポイントのフィッティングされた値、 y_i はそのポイントの測定データ値、 σ_i は y_i の標準偏差の推定値です。

もっとも単純なケースは、直線 ($y = ax + b$) にフィッティングするものです。

仮に、データが直線状に並ぶはずだという理論的な理由があるとします。

その場合は、データにもっとも適合する係数 a と b を見つけることになります。

直線または多項式関数については、1 ステップで最適な係数を見つけることができます。

これは非反復カーブフィッティングであり、多項式フィッティングに特異値分解アルゴリズムを使っています。

反復フィッティング

その他の組み込みのフィッティング関数およびユーザー定義関数については、フィッティングが未知の係数をさまざまな値を試すため、処理は反復的になります。

試行ごとに、カイ二乗値を計算し、カイ二乗値の最小値を与える係数の値を探索します。

カイ二乗値を最小化する係数の値を探索するために、レベンバーグ・マルカート（Levenberg-Marquardt）アルゴリズムが使われます。

これは非線形最小二乗近似の一種です。

フィッティングが進み、より良い値が見つかり、カイ二乗値は減少していきます。

カイ二乗値の減少率が十分に小さくなった時点で、フィッティングは終了します。

反復カーブフィッティング中は、カーブフィッティングの進捗状況を示すウィンドウが表示されます。

これは、適用された関数、更新された係数の値、カイ二乗値、パス数を示します。

通常、Quit ボタンが無効になり、OK ボタンが有効になるまで、フィッティング処理が行われます。

OK をクリックすると、フィッティングの結果が履歴エリアに表示されます。

フィッティングが十分に進み、満足のいく結果が得られた場合は、Quit ボタンをクリックすると、進行中の反復が終了し、フィッティングが自動的に完了したかのように結果が履歴エリアに表示されます。

カイ二乗値が減少しない場合や、係数の一部が非常に大きな無意味な値をとるなど、フィッティングがうまくいっていないことがわかる場合もあります。

ヘルプ User Abort Key Combinations で説明しているキーの組み合わせを押すと、フィッティングの結果を破棄してキャンセルできます。

この場合、フィッティング係数を調整して、再度試す必要があります。

初期推定

カイ二乗の最小値を探索するために、レベンバーグ・マルカート・アルゴリズムが使われます。

カイ二乗は、多次元誤差空間におけるサーフェスを定義します。

探索プロセスでは、最初に係数値の推定から開始します。

初期推定値から開始し、カイ二乗サーフェス上の出発点から丘を下るように移動しながら、最小値を探索します。

カイ二乗サーフェスで最も深い谷を見つけるのが目標です。

これは、実験データとフィッティングデータの差を最小二乗法の観点で最小化するフィッティング関数の係数の値のサーフェス上のポイントです。

フィッティング関数のうち、谷が 1 つしかないものもあります。

この場合、谷底が見つかった時点で、最もフィットする値が得られたことになります。

しかし、一部の関数では、複数の谷（周囲の値よりもフィッティングが良好な場所）が存在する可能性があるため、それが最適なフィットであるとは限りません。

フィッティング処理が谷底を見つけると、たとえサーフェスの他の場所にさらに深い谷があるとしても、フィッティングが完了したと判断します。

どの谷が最初に見つかるかは、初期推定によって決まります。

組み込みのフィッティング関数については、初期推定値を自動的に設定することができます。
これで満足のいく結果が得られない場合は、ユーザーが推定して手動で設定することになります。
ユーザー定義関数にフィッティングさせるには、手動で推定値を入力する必要があります。

終了基準

カーブフィッティングは、最適なフィットを探索するための 40 回のパスを行った後に終了しますが、9 回連続でカイ二乗値の減少がみられない場合は、その時点で終了します。

これは、初期推定が非常に正確で、最小のカイ二乗値からフィッティングが始まる場合に起こりえます。
初期推定が大幅に外れている場合や、関数がデータに全くフィットしない場合にも、このようなことが起こります。

40 回のパスの条件は変更できます。

「カーブフィッティングのための特別な変数」のセクションの `V_FitMaxIters` の説明を参照してください。

通常、40 回以上のパスが必要な場合は、フィッティングに問題があることを示しています。

「識別可能性の問題」のセクションを参照してください。

フィッティング関数とデータについてよく知らない限り、結果が良いものであると仮定するのは賢明ではありません。

ほとんどの場合、結果とデータを比較するために、結果のグラフを表示したいと思うでしょう。

残差、つまりフィッティングされたモデルとデータの差のグラフも見ることを勧めます。

Igor では、ほとんどの場合、どちらも簡単に実行できます。

カーブフィッティングでのエラー

場合によっては、カイ二乗最小値を求めるために次にどこへ進むかを決定できない状況に遭遇することがあります。

これは「特異行列」エラーの原因となります。

これについては、「カーブフィッティングにおける特異点」のセクションで説明しています。

「カーブフィッティングのトラブルシューティング」のセクションを参照すると、問題の解決策を見つけることができます。

カーブフィッティングのためのデータ

従属変数（通常「 y 」と呼ばれる）と独立変数（通常「 x 」と呼ばれる）の両方の値を測定している必要があります（特に、独立変数が 1 つしかない場合）。

これらは「応答変数」や「説明変数」と呼ばれることがあります。

ウェーブフォームデータまたは XY データに対してカーブフィッティングを行うことができます。

つまり、1 つのウェーブに含まれるデータをフィッティングすることができ、そのウェーブのデータ値は Y データを表し、 X スケーリングは等間隔の X データを表します。

または、あるウェーブのデータ値が Y 値を表し、別のウェーブのデータ値が X 値を表すような、2 つの（またはそれ以上の）ウェーブのデータをフィッティングすることもできます。

この場合、データは等間隔である必要はありません。

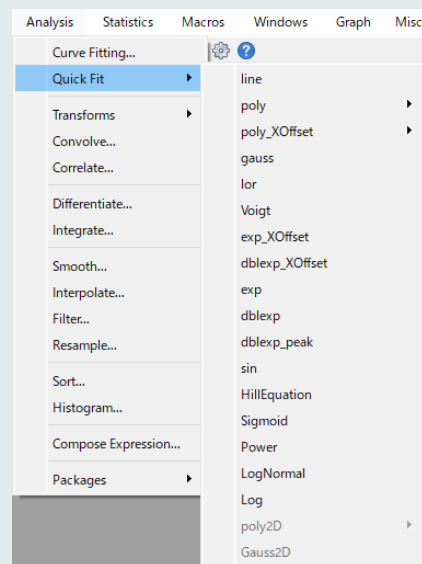
実際、 X データはランダムな順序でも構いません。

ウェーブフォームと XY データの詳細については、ヘルプ Waves を参照してください。

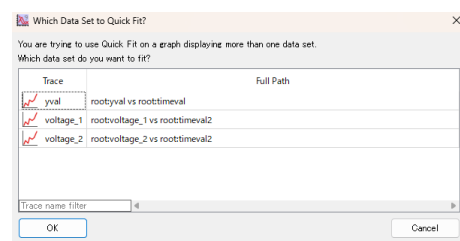
Quick Fit メニューを使ったカーブフィッティング

メニュー Analysis→Quick Fit が、カーブフィッティングを行うもっとも簡単な方法です。

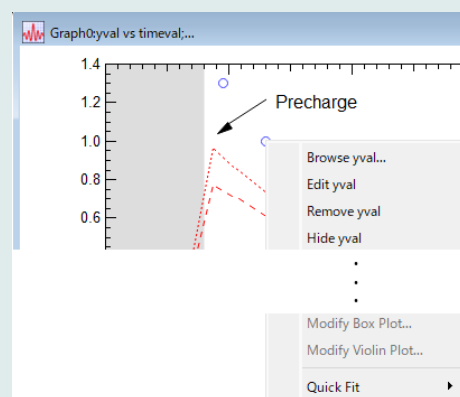
Quick Fit メニューでは、ビルトインのフィッティング関数を使って、カーブフィッティングに素早くアクセスできます。
フィッティングするデータは、最前面のグラフを調べることで決定されます。



1つのトレースだけが見つかった場合、グラフ化されたデータは選択されたフィッティング関数でフィッティングされます。
グラフに複数のトレースが含まれる場合、どのトレースをフィッティングするかを選択するダイアログが表示されます。



グラフのコンテキストメニューからも Quick Fit メニューにアクセスできます。
グラフのトレースを右クリックすると、コンテキストメニューに下のほうに Quick Fit の項目があります。
このようにして Quick Fit メニューにアクセスすると、クリックしたトレースを自動的にフィッティングします。
これにより、グラフ上に複数のトレースがある場合に、どのトレースに Quick Fit を使うのかを指定するダイアログを回避することができます。



Quick Fit メニューを使うと、フィッティングを実行し、モデルカーブをグラフに自動的に追加するコマンドが生成されます。

デフォルトでは、グラフカーソルが存在する場合、カーソル間のデータのみがフィッティングされます。

Quick Fit メニューの Fit Between Cursors を選択して、そのチェックを外すことで、データセット全体にフィッティングすることができます。

チェックされていない場合、グラフのカーソルを無視してフィッティングが実行されます。

フィッティングするトレースにエラーバーがあり、エラーバーのデータがウェーブから取得されている場合、Quick Fit はフィッティングの重みづけウェーブとしてそのウェーブを使います。

これは、エラーバーが1標準偏差を表していることを前提としていることに注意してください。

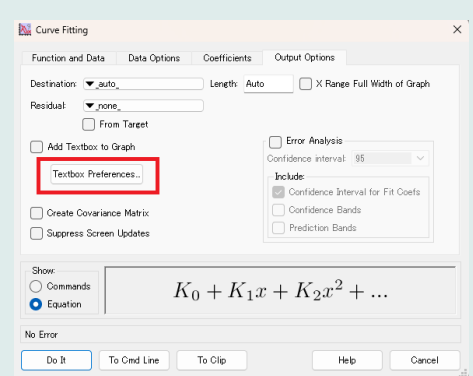
もし、エラーウェーブが1標準偏差以上を表している場合、または信頼区間を表している場合は、重み付けに使うべきではありません。

Quick Fit メニューの Weight from Error bar Wave を選択してチェックを外すと、エラーバーウェーブを重み付けに使うのを防ぐことができます。

デフォルトでは、カーブフィッティングの結果のレポートが履歴エリアに表示されます。

Textbox Preferences ボタンをクリックすると、Curve Fit Textbox Preferences ダイアログが表示されます。

設定は、メニュー Analysis → Curve Fitting を選択して、Curve Fitting ダイアログで Textbox Preferences ボタンをクリックします。



Curve Fit Textbox Preferences ダイアログで、Display Curve Fit Info Textbox をチェックし、下のリストから表示したい項目を選択します。

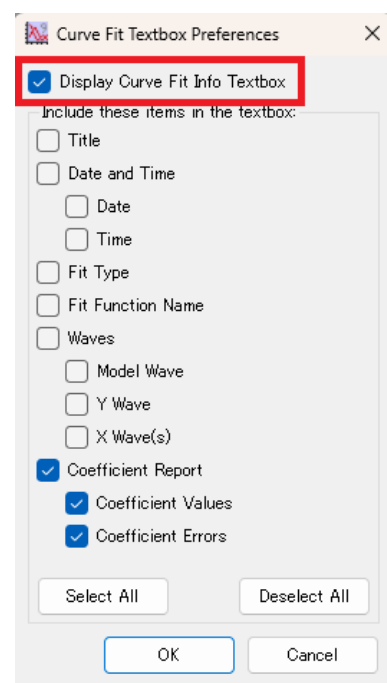
履歴エリアに出力される情報のほとんどを含むテキストボックスをグラフに追加するように指定することができます。

Analysis メニューでは、poly2D と Gauss2D のフィッティング関数が使えなくなっていることがあります。

これは、グラフに等高線プロットまたは画像プロットが含まれていないためです。

これらのプロットが含まれている場合は、これらのフィッティング関数を使うことができます。

組み込みのフィッティング関数については、「組み込みのフィッティング関数」のセクションを参照してください。



Quick Fit メニューの制限

Quick Fit メニューでは、使うことができるすべてのカーブフィッティングオプションにアクセスすることはできません。

ユーザー定義のフィッティング関数、自動残差計算、マスキング、信頼区間分析にはアクセスできません。

Quick Fit では、常に自動推定が使われます。

自動推定がうまくいかない場合は、Curve Fitting ダイアログを使って手動で推定を入力する必要があります。

グラフに、画像ピクセルサイズを設定するために補助的な X および Y ウェーブを使う画像が表示されている場合、Quick Fit ではフィッティングすることができません。

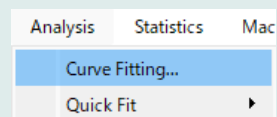
これは、画像プロット用のウェーブが余分なポイントを持つため、フィッティングに適していないためです。

コンタープロットでは、データの中心となる X と Y のウェーブを使っていて、これらはフィッティングに使うことができます。

Quick Fit は、このような等高線に対しては、適切な処理を行います。

Curve Fitting ダイアログを使ったカーブフィッティング

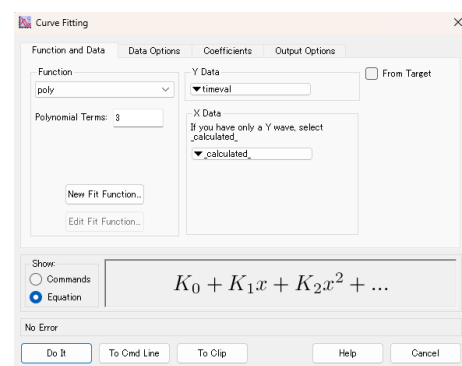
Quick Fit メニューで使うことができないオプションが必要な場合、その次に簡単なフィッティングの方法は、Analysis メニューから Curve Fitting を選択することです。



これは、Curve Fitting ダイアログを表示し、フィッティング関数とデータウェーブの選択、およびカーブフィッティングの各種オプションの設定を行うことができます。

必要に応じて、ダイアログで初期推定を入力することができます。

Curve Fitting ダイアログは、新しいユーザー定義フィッティング関数を作るためにも使えます。



ほとんどのカーブフィッティングは、Curve Fitting ダイアログを使って行うことができます。

同じフィッティング関数を使って、多数のデータセットに多数のフィッティングを行う必要がある場合、おそらく Igor のプログラミング言語でプロシージャを記述して行うことになるでしょう。

Curve Fitting ダイアログを使ってユーザー定義フィッティング関数を作る機能は、ほとんどの一般的なケースに対応できますが、非常に複雑なフィッティング関数を作るには最適な方法ではないかもしれません。

そのような場合、プロシージャウィンドウにフィッティング関数を記述する必要があります。

これは、「ユーザー定義フィッティング関数」のセクションで説明しています。

非常に複雑なユーザー定義フィッティング関数は、Curve Fitting ダイアログではうまく動作しない場合があります。

場合によっては、プロシージャウィンドウにフィッティング関数を記述し、このダイアログを使ってフィッティングの設定と実行を行う必要があります。

その他のケースでは、ユーザープロシージャを使うか、コマンドラインで入力して、手動で操作を行う必要があるかもしれません。

このようなケースはごく稀です。

シンプルなケース：組み込み関数を使ってフィッティング（Line Fit）

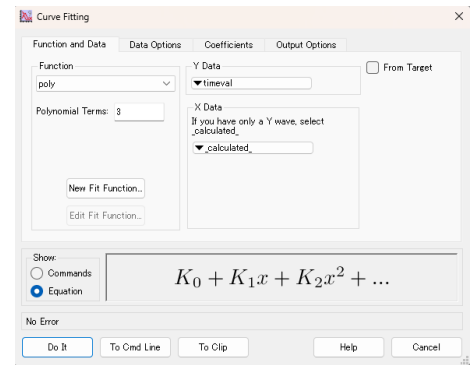
まず初めに、簡単な組み込みフィッティングである、Line Fit（直線フィッティング）について説明します。

理論的に、データは $y=ax+b$ という関数で表されるはずだと考える理由があるかもしれません。

単に、データが線に沿って並んでいるように見える、という経験則的な観察があり、その線の特徴づけたいだけなのかもしれません。

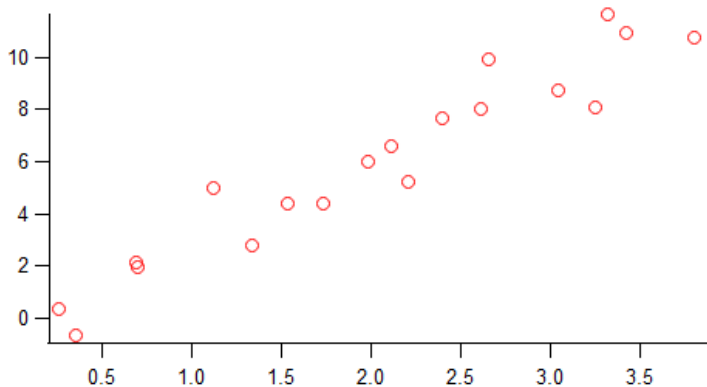
理論的な根拠あればよいのですが、そううまくはいきません。

Curve Fitting ダイアログは、4つのタブで構成されています。各タブには、フィッティング処理のいくつかの側面のコントロールが含まれています。デフォルトのオプションを使って組み込み関数にフィッティングするだけの場合は、Function タブと Data タブのみが必要となります。



グラフに表示されたデータに直線を当てはめるために必要な手順を説明します。その他の組み込み関数も、ほぼ同様の方法で動作します。

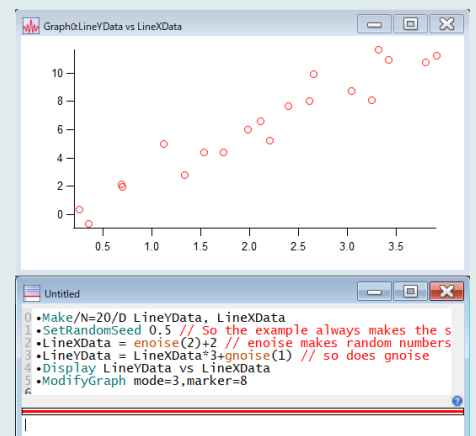
次のようなグラフでデータが表示されているとします。今、このデータに最もフィットした線を見つけたいと考えています。



1. 新しい Experiment を作成して、コマンドウィンドウで次を実行して、ウェーブとグラフを作成します。

enose 関数で生成された「ランダム」は散布データが上記と同じになるように、SetRandomSeed コマンドを使います（これを読みながら自分で操作する場合は、ここに表示されている同じデータとグラフを作成できます）。

```
Make/N=20/D LineYData, LineXData
SetRandomSeed 0.5
LineXData = enose(2)+2
LineYData = LineXData*3+gnoise(1)
Display LineYData vs LineXData
ModifyGraph mode=3,marker=8
```

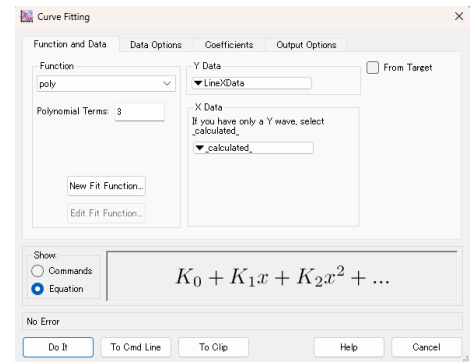


最初の行では、データを読み込む2つのウェーブを作ります。2行目では、疑似乱数ジェネレーターのシードを設定し、再現可能なノイズを生成します。3行目は、ゼロから4までの範囲で X ウェーブに一様に分布する乱数を埋めています。4行目では、原点を通り、傾きが3の直線に乗ったデータで Y ウェーブを埋め、正規分布するノイズを追加します。最後の2行でグラフを作成し、マーカーモードで表示し、マーカーとして白抜きのを設定しています。

関数とデータの選択

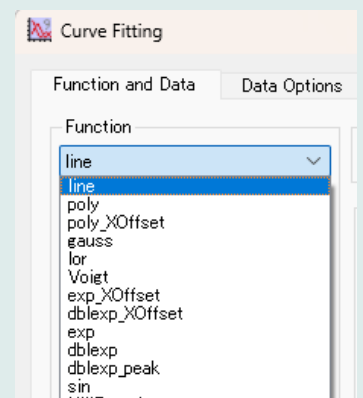
2. Analysis メニューから Curve Fitting を選択すると、Curve Fitting ダイアログが表示されます。

以前にこのダイアログを使ったことがなければ、Function and Data タブが表示された状態で画面が表示されます。



3. カーブフィッティングを行う最初のステップは、フィッティング関数を選択することです。

今回は単純な直線フィッティングを行うので、Function メニューから「line」を選択します。



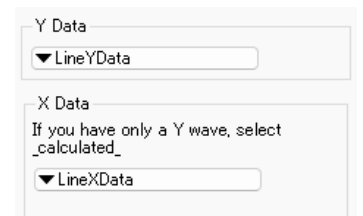
4. Y Data メニューから Y データを選択します。

ウェブフォームデータがある場合は、X Data メニューで「_calculated_」が選択されていることを確認してください。

X と Y のデータウェーブが別々になっている場合は、X Data メニューで X ウェーブを選択する必要があります。

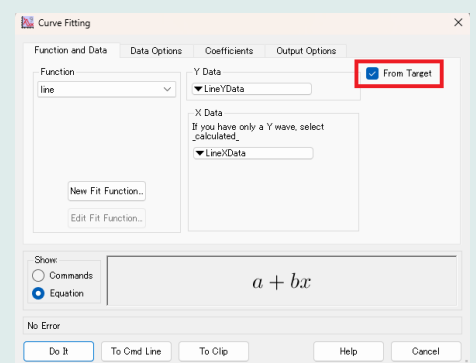
このメニューでは、Y ウェーブと同じデータポイント数を持つウェーブのみが表示されます。

メニューに X ウェーブが表示されない場合、通常はポイント数の不一致が原因です。



今回の直線フィッティングの例では、Y Data メニューから LineYData を、X Data メニューから LineXData を選択します。

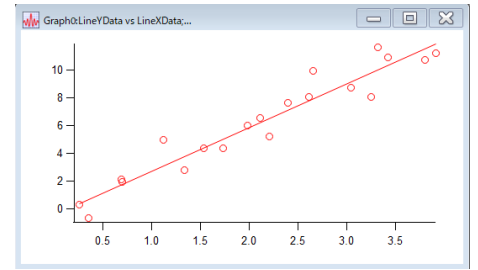
5. Experiment に多数のウェーブがある場合は、From Target チェックボックスを選択すると簡単になる場合があります。最前面のグラフまたはテーブルからウェーブのみが選択されると、Y ウェーブと X ウェーブのメニューに表示され、グラフ上のトレースで使われているウェーブのペアを選択しようと試みます。



6. これで、フィッティングを行うための準備が整いました。
 今回の単純なケースでは、ダイアログ内の他のタブを設定する必要はありません。

Do It をクリックすると、フィッティングが実行されます。

回帰フィッティングのグラフは右のような形になります。



グラフに表示されているモデルラインに加えて、履歴エリアにはさまざまな情報が表示されます。

```
•CurveFit line LineYData /X=LineXData /D ← ダイアログが生成したコマンドライン
  fit_LineYData= W_coef[0]+W_coef[1]*x ← コピーされ、モデルカーブを再評価するのに使われる
  W_coef={-0.39815,3.136} ← (ウェーブとして) フィッティング係数
  V_chisq= 20.3201;V_npnts= 20;V_numNaNs= 0;V_numINFs= 0;
  V_startRow= 0;V_endRow= 19;V_q= 1;V_Rab= -0.885758;
  V_Pr= 0.960731;V_r2= 0.923004;
  W_sigma={0.512,0.213} ← (ウェーブとして) フィッティング係数の標準偏差
  Coefficient values ± one standard deviation
      a      = -0.39815 ± 0.512 ← ダイアログに表示される名前を使って、
      b      = 3.136 ± 0.213 ← 係数値をリスト
```

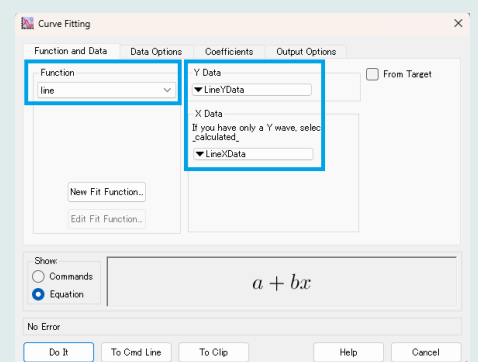
2つの便利な追加機能：係数の保持と残差の生成

フィッティングを行い、グラフを確認した結果、原点を通るべきだと考える理由があることに気付いたとします。
 測定された Y 値にばらつきがあるため、フィッティングされた直線は原点を外れています。
 解決策は、Y 切片係数をゼロに固定した状態で、フィッティングを再度行うことです。

また、フィッティングの視覚的な確認として残差を表示したい場合もあるでしょう。

7. Curve Fitting ダイアログを再度表示します。

ダイアログは前回使用した設定を記憶しているので、Function メニューでは line フィット関数が既に選択され、Y Data メニューと X Data メニューではデータウェーブが選択されています。



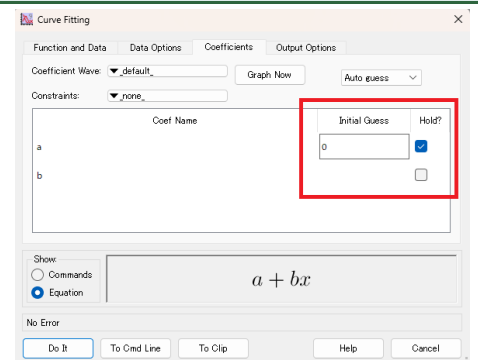
8. Coefficients タブを選択します。

各係数は係数リストの行に対応しています。

「Hold?」と表示された列のチェックボックスをクリックすると、その係数の値が固定されます。

係数の値を指定するには、Initial Guess 列にあるボックスに数値を入力します。

Hold ボックスにチェックを入れるまでは、Initial Guess ボックス



スは表示されません。

組み込みのフィッティングには初期推定値が不要なためです。

値を入力するにはボックスをクリックします。

入力が完了したら、Enter キーを押してボックスの編集モードを終了します。

9. 次に、フィッティングの残差を計算し、グラフに追加します。

Output Options タブをクリックし、**Residual** メニューから **_auto trace_** を選択します。

残差にはいくつかのオプションがあります。

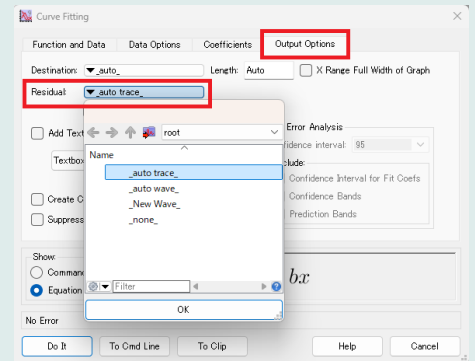
ここでは、残差を計算してグラフに追加するため、**_auto trace_** を選択しました。

常にグラフに残差を追加したいわけではないかもしれません。

auto wave を選択すると、残差は自動的に計算されますが、グラフには表示されません。

auto trace と **_auto wave_** のどちらも、Y ウェーブと同じ名前のウェーブを作成しますが、名前には「Res_」がプリフィックスとして付けられます。

New Wave を選択すると、新しいウェーブを作成するコマンドが生成され、任意の名前を付けて残差で満たすことができます。グラフには追加されません。

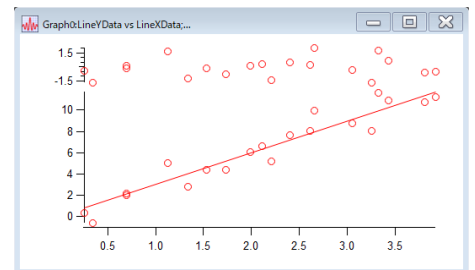


10. **Do It** をクリックすると、直線が原点を通るようにフィッティングが再計算され、ゼロに固定されます。

残差も計算され、グラフに追加されます。

グラフの線がゼロで垂直軸と交差していないことに注意してください。

これは、横軸がゼロまで伸びていないためです。



a をゼロに固定すると、履歴エリアに表示されるフィッティングの結果は次のようになります。

CurveFit コマンドの /H フラグは 1 つ以上の係数が固定されていることを示します。

```
•K0 = 0;
•CurveFit/H="10" line LineYData /X=LineXData /D /R
  fit_LineYData= W_coef[0]+W_coef[1]*x
  Res_LineYData= LineYData[p] - (W_coef[0]+W_coef[1]*LineXData[p])
  W_coef={0,2.989}
  V_chisq= 21.0032;V_npnts= 20;V_numNaNs= 0;V_numINFs= 0;
  V_startRow= 0;V_endRow= 19;V_q= 1;V_Rab= 0;V_Pr= 0.960731;
  V_r2= 0.839008;
  W_sigma={0,0.0981}
  Coefficient values ± one standard deviation
    a      = 0 ± 0
    b      = 2.989 ± 0.0981
```

← a は固定されているためゼロ

自動推定が動作しない場合

ほとんどの組み込みフィッティングは Line Fit と同じように機能します。

Function メニューから機能を選択し、データウェーブ (X 軸と Y 軸の両方のウェーブがある場合は両方) を選択し、Output Options タブで出力オプションを選択するだけです。

組み込みフィッティングでは、係数を保持したい場合を除いて、Coefficients タブは必要ありません。

しかし、自動推定が機能しない場合があります。

その場合は、Coefficients タブを使って、初期推定を設定する必要があります。

このことが当てはまる重要なケースの 1 つは、 $y=ae^{bx}$ (b は正) のような指数関数的に大きくなるデータにフィッティングしようとしている場合です。

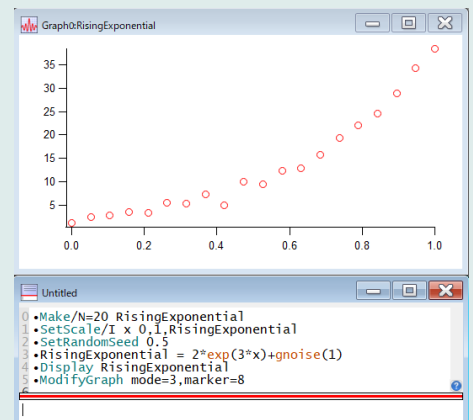
このセクションの例を確認するための手順を以下に示します。

1. 新しい Experiment を作成して、コマンドウィンドウで次を実行して、ウェーブとグラフを作成します。

コマンドラインでこれらのコマンドを 1 つずつ入力して、操作を確認することもできます。

```
Make/N=20 RisingExponential
SetScale/I x 0,1,RisingExponential
SetRandomSeed 0.5
RisingExponential = 2*exp(3*x)+gnoise(1)
Display RisingExponential
ModifyGraph mode=3,marker=8
```

これは、20 ポイントのウェーブを作成し、その X スケーリングを 0 から 1 の範囲をカバーするように設定し、指数値とノイズを少し加えてウェーブを満たし、グラフを作成します。



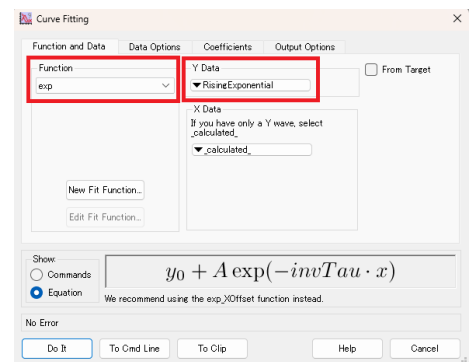
2. メニュー Analysis → Curve Fitting を選択します。

指数関数にフィッティングする最初の試みは、Function メニューから exp を選択し、Y Data メニューから RisingExponential ウェーブを選択することです。

前のセクションから続いている場合は、Coefficients タブで y_0 係数のロックを解除し、Output Options タブで Residual メニューの `_auto trace_` の選択を解除する必要があるかもしれません。

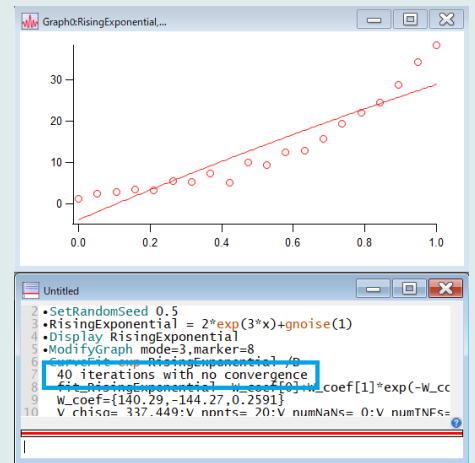
Do It をクリックします。

Fit Progress ダイアログが表示されるので、**OK** をクリックします。



3. 自動推定は、指数データが指数の負の係数によってうまく説明されると仮定しているため、フィッティングはうまくいきません。

```
•CurveFit exp RisingExponential /D
40 iterations with no convergence
fit_RisingExponential= W_coef[0]+W_coef[1]*exp(-
W_coef[2]*x)
W_coef={140.29,-144.27,0.2591}
V_chisq= 337.449;V_npnts= 20;V_numNaNs=
0;V_numINFs= 0;
V_startRow= 0;V_endRow= 19;
W_sigma={362,360,0.739}
Coefficient values ± one standard deviation
y0      = 140.29 ± 362
A       = -144.27 ± 360
invTau  = 0.2591 ± 0.739
```



グラフ化されたフィッティング曲線がデータポイントに沿っていないという事実に加えて、フィッティング係数の推定不確実性が異常に大きいです。

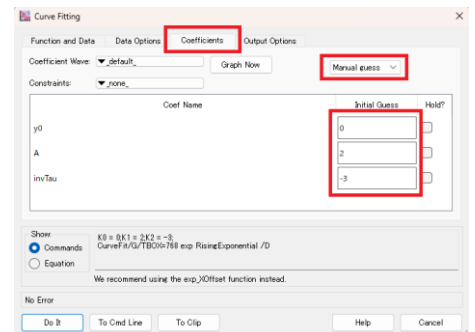
4. 解決策は、独自の初期推定値を入力することです。

Curve Fitting ダイアログの Coefficients タブをクリックし、右上のメニューで Manual Guesses を選択します。

係数リストの Initial Guess 欄に、invTau の負の値を含む初期推定値を入力できるようになりました。

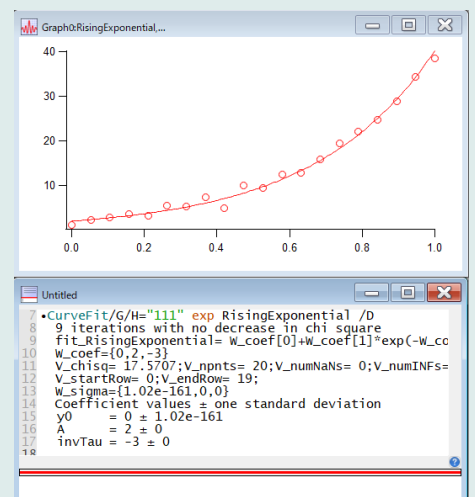
このケースでは、次のような初期推定値を入力します：

```
y0      0
A       2
invTau  -3
```



5. Do It をクリックすると、初期推定値を設定するための追加のコマンドを生成し、今後はフィッティングが正しく機能します。

```
•K0 = 0;K1 = 2;K2 = -3;
•CurveFit/G/TBOX=768 exp RisingExponential /D
Fit converged properly
fit_RisingExponential= W_coef[0]+W_coef[1]*exp(-
W_coef[2]*x)
W_coef={-0.95257,2.6625,-2.707}
V_chisq= 14.1036;V_npnts= 20;V_numNaNs= 0;V_numINFs=
0;
V_startRow= 0;V_endRow= 19;
W_sigma={0.832,0.461,0.164}
Coefficient values ± one standard deviation
y0      = -0.95257 ± 0.832
A       = 2.6625 ± 0.461
invTau  = -2.707 ± 0.164
```



ゼロから初期推定値のセットを見つけるのは難しいかもしれません。

自動推定は、修正することで適切な初期推定値を提供する良い出発点となる可能性があります。

そのため、ダイアログには Only Guess モードが用意されています。

Only Guess が選択されている場合、自動初期推定値を作成するために Do It をクリックするとフィッティングを行わずに終了します。

もう一度、Curve Fitting ダイアログを開くと、自動推定によって作成された係数ウェーブ（Coefficient Wave メニューで _default_ を選択した場合は W_coef）を選択できるようになります。

このウェーブを選択すると、初期推定値が自動推定値として設定されます。

次に、Manual Guess を選択し、初期推定を修正します。

Graph Now ボタンは、良い初期推定値を見つけるのに役立つかもしれませんが（「ユーザー定義関数の Coefficient タブ」のセクションを参照）。

定数でフィッティング

組み込みフィッティング関数のうちのいくつかは、フィッティング中に変化しない定数を含んでいます。

これらは、例えば、数値の安定性を向上させるための一定の X オフセットを提供するためにのみ入力します。

このような組み込みフィッティング関数の 1 つに exp_XOffset フィッティング関数があります。

これは次の式です。

$$y_0 + A \exp\left(\frac{x - x_0}{\tau}\right)$$

ここで、 y_0 、 A 、 τ はフィッティング係数であり、反復的なフィッティングの過程で変化し、最終的な値がフィッティングの解となります。

一方、 x_0 は定数であり、変化することはありません。

むしろ、フィッティングの設定の一部として任意の値を設定します。

exp_Xoffset フィッティング関数の場合は、ユーザーが設定しないと、デフォルトで入力データの最小の X 値に設定されます。

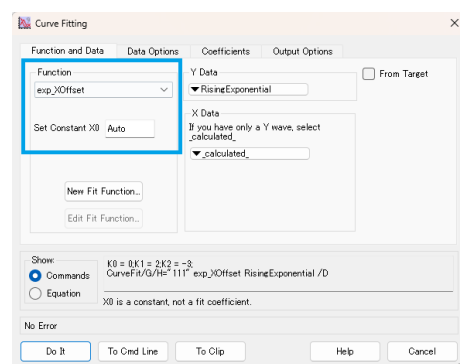
原点から離れたデータにフィッティングする場合、数値安定性が向上します。

当然、最終的な解の A の値に影響します。

Curve Fitting ダイアログで、定数を使う組み込みフィッティング関数を選択すると、フィッティング関数メニューの下に追加の編集ボックスが表示され、定数の値を設定できるようになります。デフォルト値の Auto に設定すると、入力データに基づいて定数を妥当な値に設定します。

定数を使う組み込みフィッティング関数を使ってフィッティングを行うと、出力に W_fitConstants というウェーブが含まれます。このウェーブの各要素は、方程式の定数の値を保持しています。複数の定数を使う組み込みフィッティング関数ではない場合、ウェーブには要素が 1 つだけ含まれます。

特定のフィッティング関数で使われる定数については、「組み込みフィッティング関数」のセクションを参照してください。



ユーザー定義関数でフィッティング

ユーザー定義関数へのフィッティングは、組み込み関数へのフィッティングとほぼ同じですが、主に 2 つの違いがあります：

- フィッティング関数を定義する必要がある
- 初期推定値を指定する必要がある

ユーザー定義フィッティング関数の作成を説明するために、対数関数をフィッティングする関数を作成します： $y = C_1 + C_2 \ln(x)$ 。

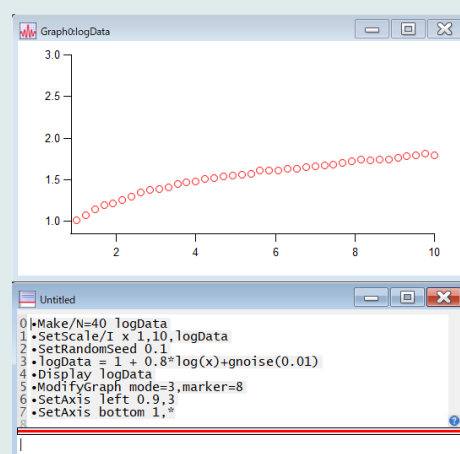
関数の作成

1. 新しいエクスペリメントを作成して、コマンドウィンドウで次を実行して、ウェーブとグラフを作成します。

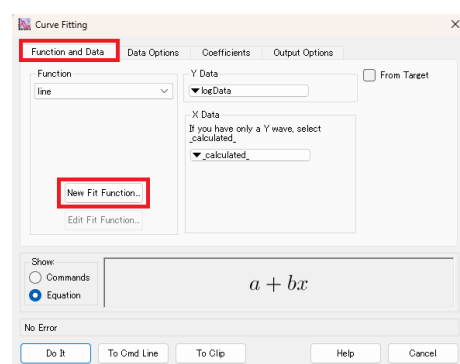
(ここでは、対数データの生成→グラフの作成→マーカーと軸の設定の順で行いました)

```
Make/N=40 logData
SetScale/I x 1,10,logData
SetRandomSeed 0.1
logData = 1 + 0.8*log(x)+gnoise(0.01)
Display logData
ModifyGraph mode=3,marker=8
SetAxis left 0.9,3
SetAxis bottom 1,*
```

これは、40 ポイントのウェーブを作成し、その X スケーリングを 1 から 10 の範囲をカバーするように設定し、対数値とノイズを少し加えてウェーブを満たし、グラフを作成します。

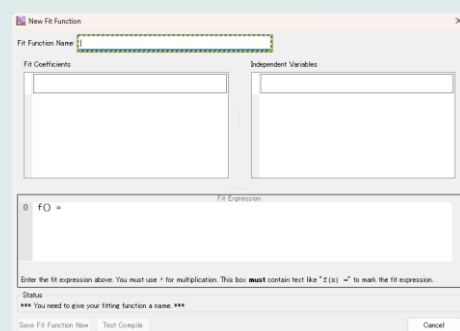


2. ユーザー定義フィッティング関数を作成するには、Curve Fitting ダイアログ (メニュー Analysis → Curve Fitting) の Function and Data タブにある New Fit Function ボタンをクリックします。



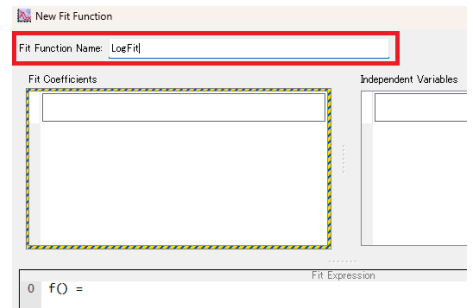
3. New Fit Function ダイアログが表示されます。

関数名を入力し、Fit Coefficients リストに係数名を入力し、Independent Variables リストに独立変数名を入力し、Fit Expression ウィンドウにフィッティング式を入力する必要があります。



4. 関数名は、非自由形式の命名規則に従う必要があります。
つまり、文字で始まり、文字、数字、アンダースコア文字のみを含み、255 バイト以下でなければなりません（ヘルプ Object Names を参照）。
組み込み関数、ユーザープロシージャ、ウェーブ名など、他のオブジェクトの名前と同じであってははいけません。

この例でのログ関数には、「LogFit」と名付けます。



5. Tab キーを押すと、フィッティング係数のリストの最初の項目に移動します。

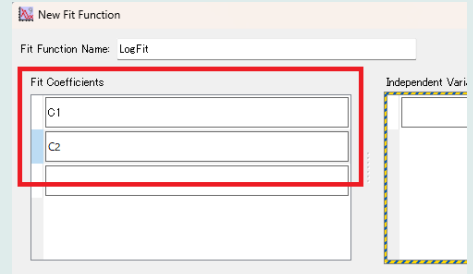
リストには常に 1 つの空欄があり、ここに新しい係数名を追加することができます。

まだ何も入力していないため、空欄は 1 つだけです。

名前を入力するたびに、Enter キーを押します。

その名前が確定すると、次の名前を入力できる新しい空欄を作成します。

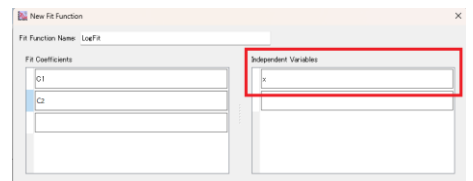
ここでは、C1 と C2 を名前として入力します。



6. Independent Variables リストの最初の空欄をクリックします。

ほとんどのフィッティング関数は、独立変数を 1 つだけ必要とします。

ここでは、独立変数に x と名前を付けることにします。



7. フィッティングの式を入力します。

Independent Variables リストに名前を入力すると、下の Fit Expression のテキストに入力されることに気づくと思います。

フィッティング関数の戻り値（ほとんどの場合 Y 値）は、

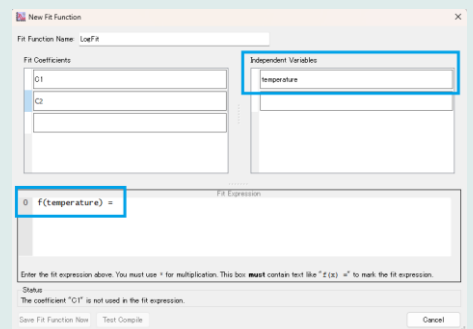
「f(x)=」 のように表示されます。

独立変数として「temperature」と入力した場合は、

「f(temperature)=」と表示されます。

この「f(x)=」というテキストは必須です。

そうしないと、関数の戻り値が不明となります。



8. Fit Expression は代数式ではありません。

コマンドラインのコマンドと同じ形式で入力する必要があります。

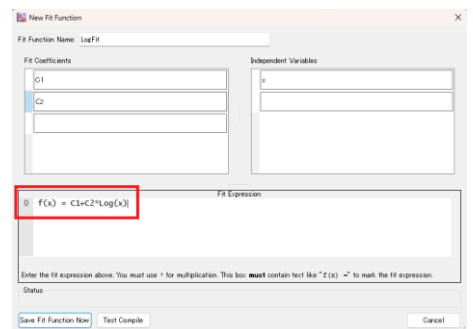
正しい記述の作成にヘルプが必要な場合は、ヘルプ Assignment Statements を参照してください。

入力する式は、代入文の右辺です。

今回の例の対数の式は次のようになります：

$$f(x) = C1 + C2 * \text{Log}(x)$$

乗算には明示的な「*」が必要です。



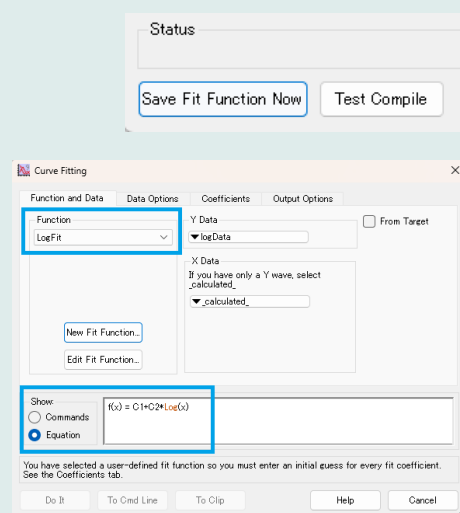
9. ダイアログは、Fit Expression のエラーをチェックします。例えば、係数または独立変数のどちらかが式から欠落している場合、Save Fit Function Now ボタンは使えません。

ダイアログでは、正しい式の構文をチェックすることはできません。

簡易的にチェックできる項目がすべて正しい場合、Save Fit Function Now と Test Compile ボタンが有効になります。

どちらかをクリックすると、プロシージャウィンドウに新しい関数が入力され、プロシージャのコンパイルが試行されます。

Save Fit Function Now ボタンをクリックしてコンパイルが成功すると、**Function** メニューで新しい関数が選択された状態で、**Curve Fitting** ダイアログに戻ります。



10. コンパイルエラーが発生した場合は、コンパイラのエラーメッセージがステータスボックスに表示され、問題のある部分がハイライト表示されます。

よくある間違いは、式のどこかで係数名をスペルミスしてしまうことです。

例えば、C1 の代わりに CC1 とタイプしてしまった場合、右のような結果になる可能性があります。

C1 が式に表示されていることに注意してください。

そうでなければ、ダイアログに C1 が欠落していることが表示されます。

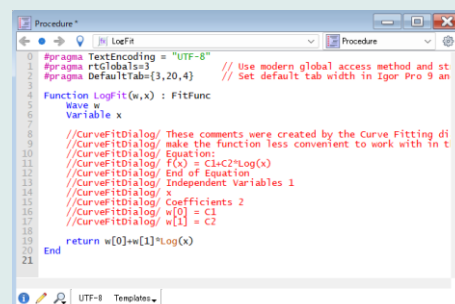


11. 準備がすべて整ったら、Save Fit Function Now ボタンをクリックして、Procedure ウィンドウに関数を構築します。Curve Fitting ダイアログは、Cancel していったん閉じます。

メニュー Windows → Procedure Windows → Procedure Window を選択します。

これには、ダイアログのさまざまな情報を識別する機能コード内のコメントも含んでいます。

今回の例では、関数は次のようになります：



```
Function LogFit(w,x) : FitFunc
```

```
Wave w
```

```
Variable x
```

```
//CurveFitDialog/ These comments were created by the Curve Fitting dialog. Altering them will
//CurveFitDialog/ make the function less convenient to work with in the Curve Fitting dialog.
//CurveFitDialog/ Equation:
//CurveFitDialog/ f(x) = C1+C2*Log(x)
//CurveFitDialog/ End of Equation
//CurveFitDialog/ Independent Variables 1
//CurveFitDialog/ x
//CurveFitDialog/ Coefficients 2
//CurveFitDialog/ w[0] = C1
//CurveFitDialog/ w[1] = C2
```

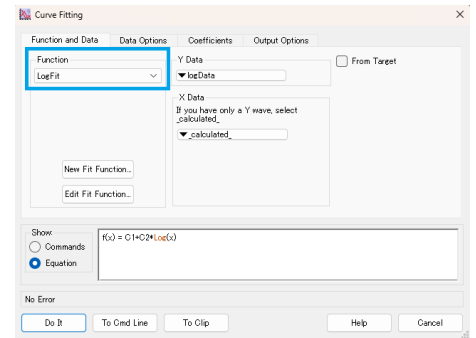
```
return w[0]+w[1]*Log(x)
```

```
End
```

関数が非常に複雑でダイアログで処理できない場合を除き、Procedure ウィンドウのコードを修正する必要はありません。

Procedure ウィンドウでのフィッティング関数の記述方法の詳細については、「ユーザー定義フィッティング関数」のセクションを参照してください。

12. もう一度 Curve Fitting ダイアログを開いて、Function メニューを確認すると、LogFit がフィッティング関数として選択できるようになっています。



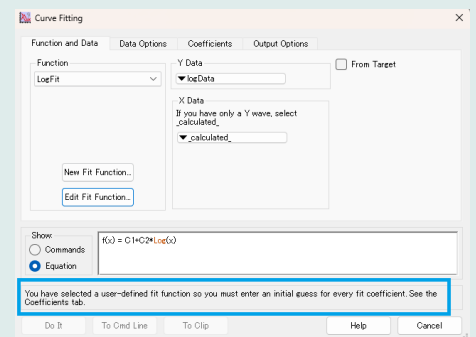
ユーザー定義関数の Coefficients タブ

ユーザー定義関数にフィッティングするには、Coefficients タブで初期推定値を入力する必要があります。

ユーザー定義のフィッティング関数を作成した（または既存の関数を選択した）場合、ダイアログの下部のエラーメッセージウィンドウに次のように表示されます：

You have selected a user-defined fit function so you must enter an initial guess for every fit coefficient. See the Coefficients tab.

（ユーザー定義のフィッティング関数を選択したため、すべてのフィッティング係数の初期推定値を入力する必要があります。Coefficients タブを見てください）

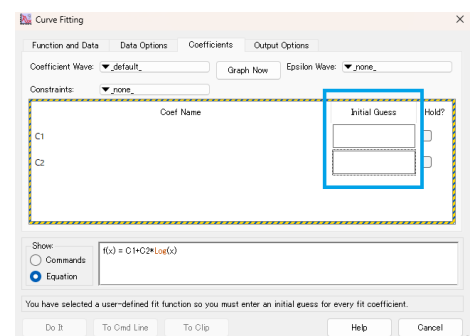


ユーザー定義のフィッティング関数を選択すると、係数リストの Initial Guess 列が有効になります。

各行には数値を入力する必要があります。

一部の関数はフィッティングするのが難しい場合があります。

そのような場合、初期推定値が最終的な解にかなり近い値になることがあります。

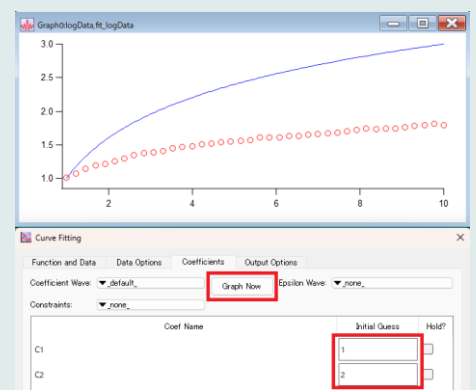


初期推定値を適切に設定できるように、Coefficients タブには Graph Now ボタンが用意されています。

このボタンをクリックすると、入力した初期推定値を使って、フィッティング関数を表すトレースが前面のグラフに追加されます。

初期推定の列の値を変更し、Graph Now ボタンを何度でもクリックすることができます。

変更が加えられるたびにトレースが更新されます。

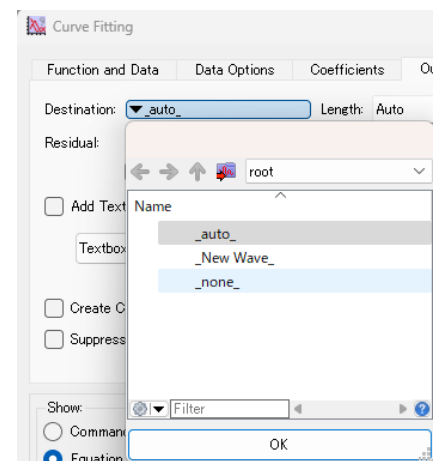


Graph Now ボタンは、「宛先ウェーブ」のセクションで説明しているように動作しますが、1つの例外があります。

Output Options タブの Destination ポップアップメニューで `_none_` を選択した場合、Graph Now ボタンは `_auto_` を選択した場合と同じように動作します。

Graph Now ボタンは、選択した Destination のウェーブスタイルを反映し、`_New Wave_` を選択した場合は新しいウェーブを作成します。

Do It をクリックする前に Destination のウェーブ設定を変更しない限り、Graph Now で設定したウェーブはフィッティングによって上書きされます。



Coefficient タブでは、「イプシロン（誤差項）」ウェーブを選択するオプションがあります。

イプシロンウェーブには、係数ウェーブの各ポイントのイプシロン値が1つずつ含まれます。

デフォルトでは、Epsilon Wave メニューは `_none_` に設定されていて、イプシロン値はデフォルト値に設定されていることを示しています。

各イプシロン値は、フィッティング係数に関する偏微分を計算するために使われます。

偏微分は、カイ二乗値が最小となる係数の探索方向を決めるために使われます。

ほとんどの場合、イプシロン値はクリティカルではありません。

ただし、デフォルトのイプシロン値が許容できる偏微分を提供していないと判断する理由がある場合は、独自の値を指定することができます（特異行列エラーを回避できる場合もあります）。

イプシロン値を指定するには、Epsilon Wave メニューから既存のウェーブを選択するか、`_New Wave_` を選択します。

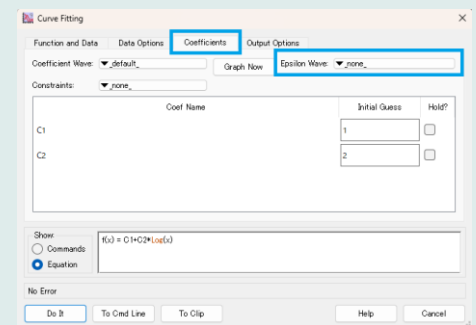
メニューには、フィッティング係数と同じポイント数を持つウェーブのみが表示されます。

どちらを選択しても、係数リストにイプシロン列が表示され、イプシロン用の値を入力することができます。

Epsilon Wave メニューからウェーブを選択すると、そのウェーブの値がリストに入力されます。

`_New Wave_` を選択すると、ダイアログがイプシロンウェーブを作成し、Epsilon 列の値でそれを埋めるためのコマンドを生成します。

イプシロンウェーブの詳細とその機能については、「イプシロンウェーブ」のセクションを参照してください。



ユーザー定義関数が常に使えるようにする

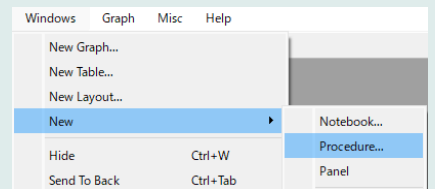
フィッティング関数は Procedure ウィンドウで作成されるため、Experiment ファイルの一部として保存されることに注意してください。

つまり、作成した Experiment で作業している間は使うことができますが、他の Experiment ファイルで作業するときは使うことができません。

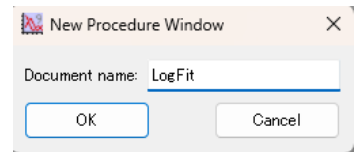
Igor Pro を起動するたびに作成したフィッティング関数を使うことができるようになります。

前のセクションからの続きで説明します。

1. Windows メニューの New から Procedure を選択して、New Procedure Window を開きます。

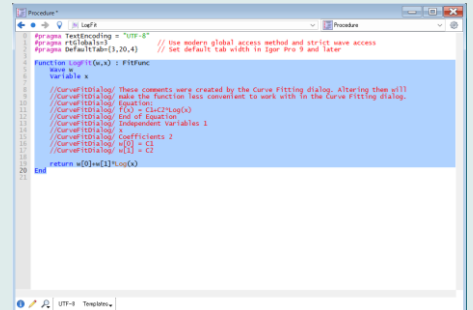


2. 前のセクションでは、フィッティング関数に LogFit と名前を付けたので、Document name には LogFit と入力して、OK をクリックします。

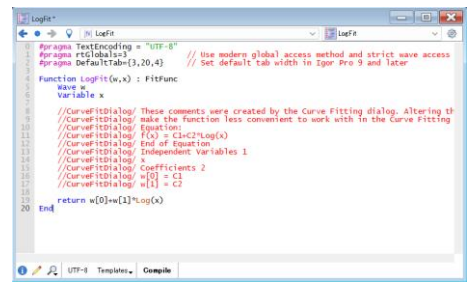


3. メニュー Windows → Procedure Windows → Procedure Window を選択します。

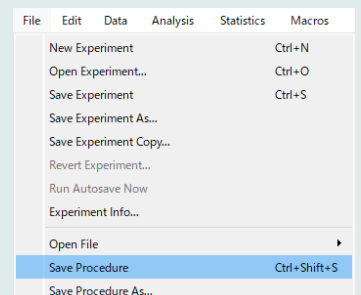
Procedure ウィンドウでフィッティング関数を見つけ、Function から End までのテキストをすべて選択し、Edit メニューから Cut を選択します。



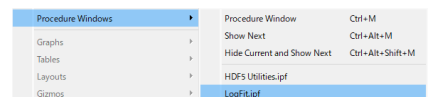
4. コードを新しいプロシージャウィンドウに貼り付けます。



5. 最後に、File メニューから Save Procedure Window を選択し、Igor Pro User Files フォルダの Igor Procedure に保存します（詳細はヘルプ Igor Pro User Files を参照）。



6. これで次に Igor Pro を起動すると、Windows メニューの Procedure Windows でそのプロシージャを選択することができます。



ユーザー定義関数を削除する

不要になったユーザー定義フィッティング関数を削除するには、Windows メニューから Procedure Windows を選択します。

Procedure ウィンドウで関数を探します（Edit メニューから Find を選択し、関数名を検索します）。

Function という語から End という語までの関数を定義するテキストをすべて選択し、テキストを削除します。

前のセクションで説明した、常に使えるように設定した場合は、Igor Pro User Files フォルダの Igor Procedures フォルダ内の対応するプロシージャファイルを見つけ、フォルダから削除して、Igor を再起動してください。

ユーザー定義フィッティング関数の詳細

ユーザー定義フィッティング関数を入力するもっとも簡単な方法は、New Fit Function ダイアログを使うことです。

しかし、フィッティングする式が非常に長い場合や、ローカル変数や条件分岐を含む複数の行が必要な場合、ダイアログが煩雑になる可能性があります。

また、特定の特殊な状況では、ダイアログでサポートされていない書式が必要になる場合があります。

ユーザー定義フィッティング関数の形式と、異なる形式の使い方についての詳細は、「ユーザー定義フィッティング関数」を参照してください。

外部関数 (XFUNC) を使ってフィッティング

外部関数 (XFUNC) とは、Igor の拡張機能またはプラグインを介して提供される関数です。

プログラマーは、XOP Toolkit を使って XFUNC を開発します。

XFUNC を使う時には、XOP Toolkit を使わずに可能です。

XFUNC を使うには事前にインストールしておく必要があります。

ヘルプ Igor Extensions を参照してください。

XFUNC は、フィッティング関数に膨大な計算が必要な場合、カーブフィッティングを大幅に高速化することができます。

通常、フィッティングの速度は、他の部分でのオーバーヘッドに依存していて、XFUNC を開発する手間をかけることは妥当ではありません。

外部関数へのフィッティングは、ユーザー定義関数へのフィッティングと同様ですが、Curve Fitting ダイアログでは必要なフィッティング係数の数を特定する手段がありません。

Coefficient タブに切り替えると、それを伝えるアラートが表示されます。

この問題の解決策は、正しいポイント数を持つ係数ウェーブを選択することです。

そのためには、Curve Fitting ダイアログを入力する前に、ウェーブを作成する必要があります。

係数ウェーブを選択すると、そのウェーブの内容が係数リストの作成に使われます。

ウェーブ値は Initial Guess 列に入力されます。

初期推定を変更すると、ダイアログがウェーブに新しい値を入力するために必要なコマンドを生成します。

通常、Coefficient Wave メニューには、フィッティング関数で必要とされるフィッティング係数の数と同じ長さのウェーブのみが表示されます。

フィッティングに XFUNC を選択すると、メニューにはすべてのウェーブが表示されます。

そのため、どれを選択するべきかを知っておく必要があります。

ウェーブ名には、そのウェーブが何のためのものかを識別できる名前を使うことをお勧めします。

Igor は、XFUNC の係数名については何もわかりません。

係数名は、選択した係数ウェーブの名前から生成します。

係数ウェーブが「coefs」と名付けられている場合、係数名は「coefs_0」「coefs_1」などとなります。

C または C++ で機能を実装する場合は、もちろん、より多くの時間がかかり、XOP Toolkit とソフトウェア開発環境の両方が必要となります。

XOP Toolkit を使って独自の外部関数を作る方法の詳細については、ヘルプ Igor Extensions を参照してください。

係数ウェーブ

ユーザー定義関数にフィッティングさせる場合、初期推定値は係数ウェーブを介してカーブフィッティング操作に伝達されます。

選択した関数の種類にかかわらず、フィッティングの結果として得られる係数は係数ウェーブとして出力されます。Curve Fitting ダイアログでは、このことがほとんど隠されています。

ユーザー定義関数を作成すると、ダイアログはフィッティング係数を含むウェーブを入力として受け取る関数を作成します。

しかし、関数コード内の特別なコメントを通じて、ダイアログは各係数に名前を付けます。

組み込み関数には、内部で保存されている係数に名前が付けられています。

これらの名前を使うことで、ダイアログは係数ウェーブの使用に伴う複雑な処理の一部をユーザーから隠すことが可能です。

カーブフィッティング後の履歴の出力では、係数値は実際の係数ウェーブを使ったウェーブの代入形式と、係数名を使ったリスト形式の両方で出力されます。

例えば、先ほどのユーザー定義関数へのフィッティングの例（ユーザー定義関数へのフィッティング）の出力結果は次の通りです：

```
•FuncFit LogFit W_coef logData /D
Fit converged properly
fit_logData= LogFit(W_coef,x)
W_coef={1.0041,0.99922}
V_chisq= 0.00282525; V_npnts= 30; V_numNaNs= 0; V_numINFs= 0;
W_sigma={0.00491,0.00679}
Coefficient values ± one standard deviation
C1 = 1.0041 ± 0.491
C2 = 0.99922 ± 0.679
```

ウェーブ代入のバージョンはコマンドラインにコピーして実行できるほか、ユーザープロシージャ内のコマンドとしても使用できます。リストバージョンはより読みやすくなっています。

Coefficients タブの Coefficient Wave メニューを使って、係数ウェーブの処理方法をコントロールします。以下のオプションが用意されています。

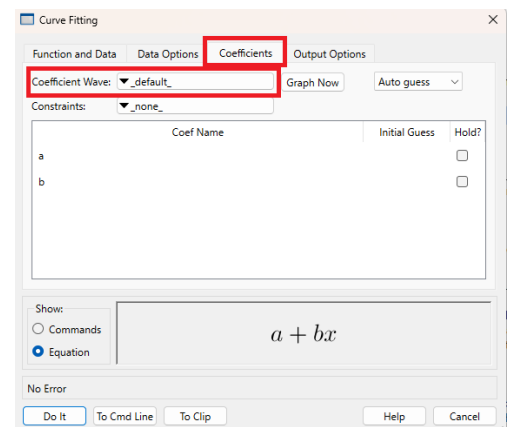
Default (デフォルト)

`_default_` を選択すると、`W_coef` というウェーブが生成されます。

組み込みのフィッティングでは、このウェーブは出力専用に使われます。

ユーザー定義のフィッティングに対しては、これも入力です。

ダイアログは、フィッティング開始前に初期推定値をウェーブに反映させるコマンドを生成します。



明示的なウェーブ

Coefficient Wave メニューには、長さがフィッティング係数の数と一致するウェーブが一覧表示されます。
これらのウェーブのいずれかを選択すると、そのウェーブがフィッティングの入力および出力に使われます。

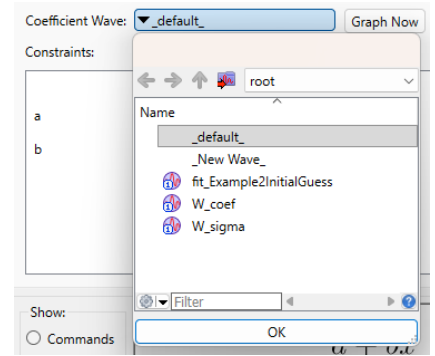
メニューからウェーブを選択すると、そのウェーブのデータが係数リストの初期推定値欄に入力されます。

これは初期推定を入力する便利な方法として利用できます。

明示的にウェーブを選択し、初期推定値を編集すると、ダイアログはフィッティング開始前に選択された係数ウェーブの値を変更するコマンドを生成します。

ウェーブの内容を変更しないようにするには、初期推定値を含むウェーブを選択した後、_default_ または _New Wave_ を選択します。

初期推定値は、新規またはデフォルトの係数ウェーブに配置されます。



New Wave (新しいウェーブ)

明示的な係数ウェーブのバリエーションとして _New Wave_ があります。

これは明示的なウェーブと同様に動作しますが、ダイアログがフィッティング開始前にウェーブを作成するコマンドを生成するため、ダイアログ入力前にウェーブを作成する手間が省けます。

ウェーブは初期推定値列の値で埋まります。

複数のフィッティングを行い、各フィッティングの係数を保存したい場合には、_New Wave_ オプションが便利です。

default を使うと、フィッティングの結果が以前のフィッティングの結果を上書きします。

誤差

フィッティング誤差の推定値（フィッティング係数の推定標準偏差）は、自動的に W_sigma という名前のウェーブに保存されます。

これに関するユーザー側の選択はありません。

宛先ウェーブ

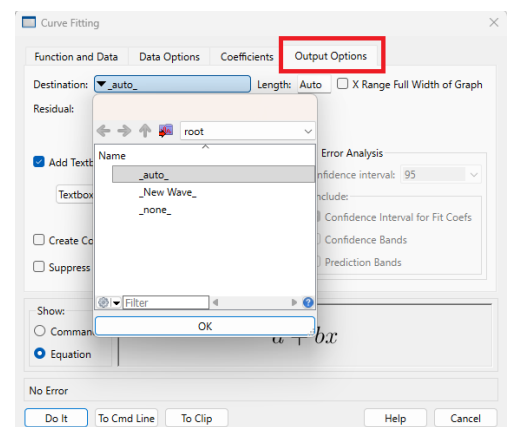
カーブフィッティングを実行すると、フィッティング係数に対応するモデルカーブが計算されます。

ほとんどの結果と同様に、モデルカーブは数値の配列としてウェーブに保存されます。

このウェーブが「宛先ウェーブ」です。

宛先ウェーブの主な目的は、フィッティング中に様々な係数を用いたフィッティング関数の様子を表示し、処理終了時には最終的なフィッティング係数を用いた関数を表示することです。

宛先ウェーブを指定しない、明示的な宛先ウェーブを選択する、または自動トレース機能を使うことができます。



ダイアログの Output Options タブで宛先ウェーブオプションを選択します。
以下のオプションがあります。

宛先なし

フィッティング処理中にグラフィカルなフィードバックを必要とせず、生データをフィッティング関数とグラフィカルに比較する必要がない場合は、宛先ウェーブを選択しないでください。
これは、大量のデータセットをバッチ処理でフィッティングする場合に該当する可能性があります。
Destination メニューで `_none_` を選択してください。

自動トレース

ほとんどの場合、自動トレースが推奨されます。

自動トレースは、Destination メニューから `_auto_` を選択することで選択します。

これを選択すると、新しいウェーブが自動的に作成され、X スケーリングが適切に設定され、Y データウェーブがグラフに表示されている場合は、そのグラフの最前面に追加されます。

新しいウェーブの名前は、Y データウェーブの名前に「fit_」をプリフィックスとして追加して生成されます。

その名前を持つウェーブが既に存在する場合、上書きされます。

ウェーブの名前が 255 文字の最大長を超える場合、名前は切り捨てられます。

宛先ウェーブにおけるポイント数は、独立変数の数に依存します。

最も一般的な単変量フィッティングの場合、デフォルトは 200 ポイントです。

以降の説明は、単変量データのフィッティングを行うことを前提としています。

自動トレースを使って同じ生データに複数の関数をフィッティングさせる場合は、後続のフィッティングによって上書きされないよう、フィッティング後に自動トレースウェーブの名前を変更してください。

Data メニューの Rename を使うか、コマンドラインから直接 Rename コマンドを実行して名前を変更できます。

同様に、W_coef ウェーブと W_sigma ウェーブも保存することをお勧めします。

通常、自動トレースのウェーブは、XY データにフィッティングする場合でもウェーブとして設定されます。

自動トレースのウェーブの X スケーリングは、200 ポイントを元のデータの X の範囲に均等に分散するように設定されます。

設定が有効な場合、グラフに追加される自動トレースのウェーブは、プリファレンスで設定された線種（色、太さなど）が適用されます。

ただし、線モードは常に「lines between points」に設定されます。

これはカーブフィッティングの結果を表示するのに最適です。

等間隔データは対数軸での曲線表示に適していません。

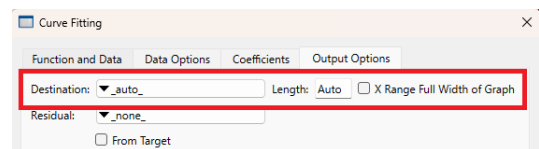
データが対数軸で表示されている場合、フィッティングにより XY ペアのウェーブが作成されます。

X ウェーブは、Y データウェーブの名前に「fitX_」を追加して命名されます。

この X ウェーブは、フィッティングデータの X 範囲全体に広がった、指数関数的に間隔が設定された X 値で埋められます。

もちろん、続いてその軸を線形軸に変更すると、ポイントの間隔は正しく見えなくなります。

Destination メニューで `_auto_` を選択すると、ダイアログに Length というラベルの付いたボックスが表示されます。これを使って、宛先ウェーブのポイント数を変更します。3 以上の任意の数値を設定できます。



ポイントが多いほど、曲線は滑らかになります（ある程度まで）。ポイントが多いほど描画に時間がかかるため、フィッティング処理は遅くなります。

明示的な宛先

自動トレースを使う代わりに、明示的な宛先ウェーブを指定できます。入力データポイントの X 位置におけるモデル値を取得したい場合に使います。

明示的な宛先ウェーブは、Y データウェーブと同じポイント数を持つ必要があるため、Duplicate コマンドを使って作成する必要があります。

Destination メニューには、選択した Y データウェーブと同じポイント数を持つウェーブのみが表示されます。

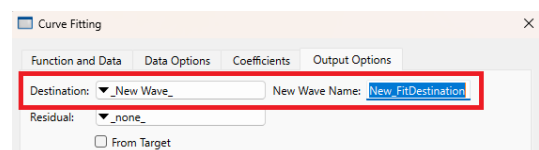
明示的な宛先ウェーブは自動的に最前面のグラフに追加されません。したがって、カーブフィッティングコマンドを実行する前に、通常は以下のようなコマンドを実行します：

```
Duplicate/O yData, yDataFit
AppendToGraph yDataFit vs xData
```

ウェーブデータを XY データではなくフィッティングする場合は、AppendToGraph コマンドで「vs xData」を省略します。

新しいウェーブ

Curve Fitting ダイアログで、Destination メニューから `_New Wave_` を選択すると、宛先ウェーブを自動的に作成できます。これは、Y データウェーブを複製するための Duplicate コマンドを生成し、それを他の明示的な宛先ウェーブと同様に使うことで実現します。



新しいウェーブはグラフに自動的に追加されないため、フィッティング完了後に手動で追加する必要があります。

データのサブセットにフィッティング

よくある問題として、データセットの全データをカーブフィッティングに含めたくない場合があります。フィッティングの対象を一部のデータに限定するには、2つの方法があります。

フィッティングする範囲を選択

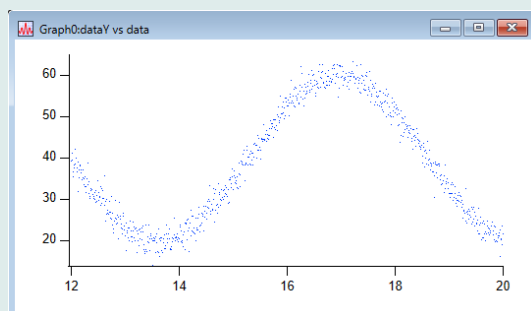
Range ボックスでは、連続したポイントの範囲を選択できます。カーブフィッティングの範囲を指定する時に使う値は、Y データウェーブのポイント番号または行番号で表します。

XY ペアのウェーブにフィッティングする場合、X 値がランダムな順序で並んでいると、グラフ上に表示される連続した範囲を選択することにはならないことに注意してください。

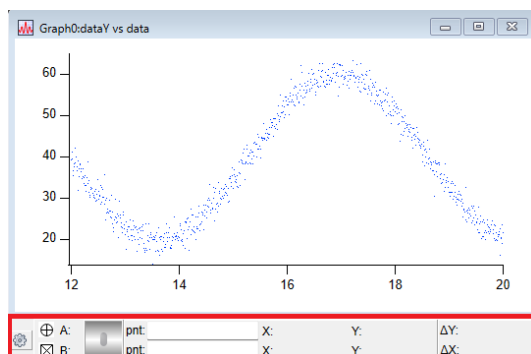
範囲の選択を簡略化するために、グラフカーソルを使って範囲の開始ポイントと終了ポイントを選択できます。カーソルを使うには、生データをグラフに表示します。

1. 例えば、次のようなデータを作って手順を確認します。

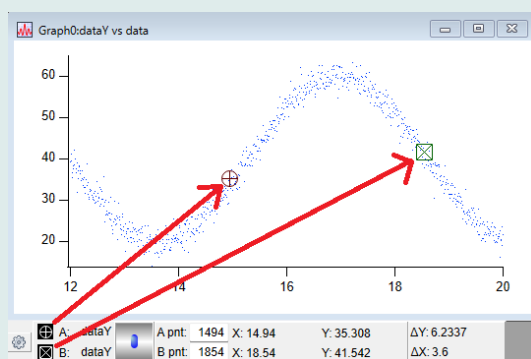
```
Make/N=3000/D data, dataY  
data=x*0.01  
dataY=gnoise(2)+20*sin((x+450)/105)+40  
Display dataY vs data  
SetAxis bottom 12,20  
ModifyGraph mode=2,rgb=(1,16019,65535)
```



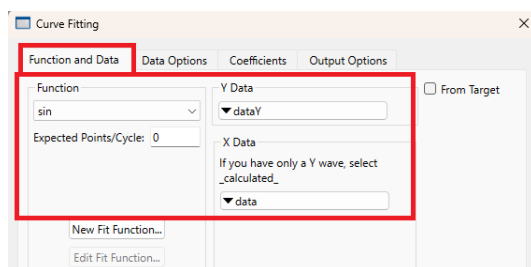
2. Graph メニューから Show Info を選択し、グラフに情報パネルを表示します。



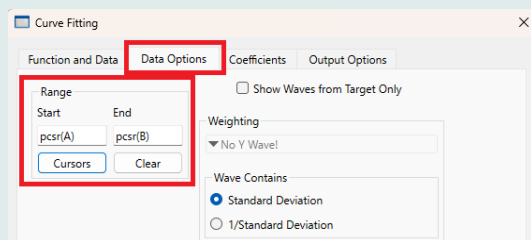
3. A と B のカーソルを元のデータ上にドラッグします。



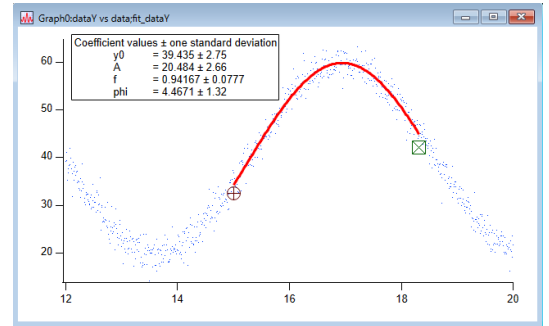
4. Analysis→Curve Fitting を選択し、Curve Fitting ダイアログ内の Function and Data タブで Function で sin (今回は Sin でフィッティング)、X/Y Data をそれぞれ選択します。



5. Data Options タブの Cursor ボタンを使って、カーソル範囲をフィッティングするコマンドを生成します。



6. データセットの一部に対してフィッティングを行った後のグラフは、右のようなものになります。
フィッティングカーブのトレースは太くしてあります。

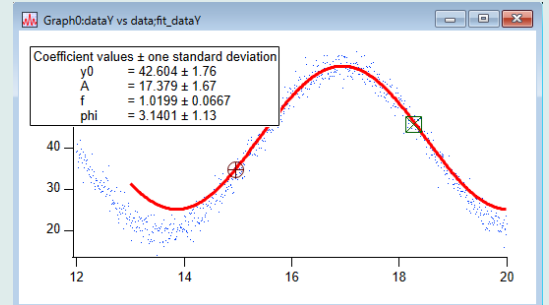


この例では、宛先に自動トレースを使用しました。
トレースは選択範囲の上にものみ表示されることに注意してください。
より広い範囲で目的のウェーブを表示したい場合、目的のウェーブの X 方向スケーリングを変更する必要があります。

7. 次のコマンドは、より広い範囲でより多くのポイントを表示するように目的のウェーブを変更します：

```
Redimension/N=500 fit_data // 500 ポイントに変更
SetScale x 13, 20, fit_data // ドメインを 13 から 20 に設定
fit_dataY= W_coef[0]+W_coef[1]*sin(W_coef[2]*x+W_coef[3])
```

最後の行は、フィッティング後に表示された履歴エリアからコピーします。
これにより、右のグラフが生成されます。



自動トレースではなく明示的な宛先ウェーブを使う場合、フィッティングを実行する前に宛先ウェーブを空白 (NaN) に設定すると効果的です。

フィッティングが進むにつれて、フィッティング対象の範囲に対応する宛先ウェーブ範囲のみに新たな値が保存されます。

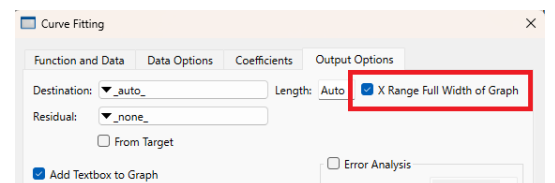
また、ソースウェーブが NaN または INF でないポイントでのみ、宛先ウェーブに格納されます。

宛先ウェーブを空白に事前に設定しない場合、宛先ウェーブには新旧のデータが混在した状態になります。

次のコマンドは、宛先ウェーブを事前に設定し、その後 XY ペアの範囲に対してカーブフィッティングを実行する方法を示しています。

```
Duplicate/O yData, yDataFit // 宛先を作成
yDataFit = NaN // 空白 (NaN) にプリセット
AppendToGraph yDataFit vs xData
CurveFit lor yData(xcsr(A),xcsr(B)) /D=yDataFit
```

フィッティングカーブの範囲を広げる別の方法は、X Range Full Width of Graph というチェックボックスを選択することです。
このチェックボックスは、Curve Fitting ダイアログの Output Options タブにあります。



マスクウェーブを使う

除外したいポイントが連続していない場合があります。

これは、時々不良なポイントを含むデータセットにフィッティングする場合に起こり得ます。

あるいは分光データでは、ベースライン領域をフィッティング対象とし、分光ピークの一部であるポイントを除外し

たい場合があります。

こういった場合は、マスクウェーブを使うことで、望ましい結果を得られます。

マスクウェーブは、Y データウェーブと同じ数のポイントを持つ必要があります。

マスクウェーブには、除外するデータポイントに対応する NaN (Not-a-Number、空白セル) またはゼロを、含めるデータポイントにはゼロ以外の値を入力します。

カーブフィッティングダイアログを表示する前に、マスクウェーブを作成する必要があります。

テーブルでマスクウェーブを編集することもできます。

テーブルに NaN を入力するには、「NaN」と入力して Enter キーを押します。

NaN を入力したら、クリップボードにコピーして他のセルに貼り付けることができます。

コマンドラインでウェーブの代入を使うこともできます。

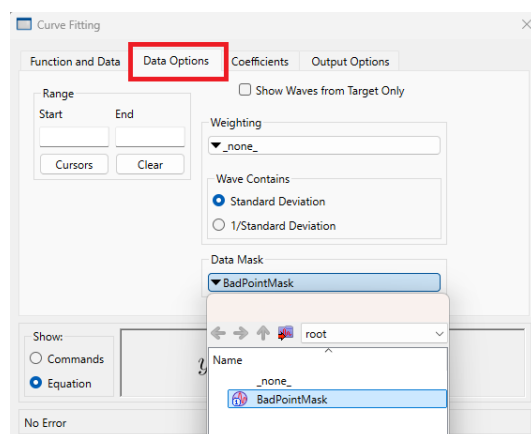
データセットのポイント 4 に不良データポイントがある場合、マスクウェーブにおいてポイント 4 を設定する適切なコマンドは次のようになります：

```
BadPointMask[4] = NaN
```

適切なマスクウェーブがある場合、Data Options タブの Data Mask メニューからそれを選択します。

マスクウェーブを Modify Trace Appearance ダイアログで f(z) ウェーブとして選択すると、NaN ポイントを含むマスクを使って、グラフ上でマスクされたポイントの表示を抑制できます。

同じウェーブを ModifyGraph mask のキーワードと組み合わせることも可能です。



重み付け

特定のデータポイントに重要度の差をつけたい場合、重み付けウェーブを指定できます。

これを行う理由は主に二つあります：

- より良く、より正確なフィッティングを得るため。
- フィッティング係数の誤差推定値をより正確に得るため。

重み付けウェーブはカイ二乗の計算に使われます。

カイ二乗は以下のように定義されます。

$$\sum_i \left(\frac{y - y_i}{w_i} \right)^2$$

ここで、y は特定のポイントに対する推定値、y_i はそのポイントの元のデータ値、w_i はそのポイントの標準誤差です。

重み付けウェーブは w_i 値を提供します。

重み付けウェーブの値は、1/σ_i または単に σ_i のいずれかです。

ここで、σ_i は各データ値の標準偏差です。

必要に応じて、Igor は重み付けを行う前に重み付け値の逆数を取ります。

重み付け値を含むウェーブは、Data Options タブの Weighting メニューから選択して指定します。

さらに、ウェーブに標準偏差が含まれるか、逆標準偏差が含まれるかを指定する必要があります。

これを行うには、メニューの下にあるボタンからいずれかを選択します：

- Standard Deviation
- 1/Standard Deviation

通常は標準偏差を使います。

逆標準偏差は歴史的な理由から認められています。

σ_i の値を得る方法はいくつかあります。

例えば、測定プロセスに関する事前知識がある場合です。

データポイントが繰り返し測定から得られた平均値である場合、適切な重み値は標準誤差となります。

これは、繰り返し測定の標準偏差を $N^{1/2}$ で割ったものです。

これは、測定誤差が平均 0 の正規分布に従うことを前提としています。

データがヒストグラムやマルチチャンネル検出器など、カウントの結果である場合、適切な重みは $\sqrt{Y}$ です。

ただし、この式ではゼロ値に対して無限大の重みが割り当てられ、これは正しくなく、それらのポイントがフィッティングから除外される原因となります。

ゼロポイントに対する重みを 1 に置き換えるのが一般的です。

特定のポイントをフィッティング処理から完全に除外するには値ゼロを使用できますが、この目的にはデータマスクウェーブを使う方が望ましいです。

重み付けウェーブを指定しない場合、フィッティングでは均等な重み付けが使われ、共分散行列はフィッティング関数がデータを適切に記述しているという仮定に基づいて正規化されます。

係数の出力誤差は共分散行列の対角要素の平方根として計算されるため、正規化処理が有効な誤差推定値を提供する条件は、全データポイントがほぼ同程度の誤差を有し、かつフィッティング関数が実際にデータに適している場合に限られます。

重み付けウェーブを指定する場合、共分散行列は正規化されず、報告される係数誤差の精度は重み付け値の精度に依存します。

このため、重み付けには任意の値を使うべきではありません。

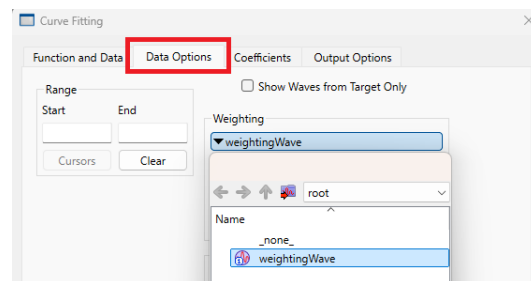
比例加重

場合によっては、重み付けを使うことが望ましいが、絶対的な測定誤差ではなく比例加重しか知らないことがあります。

この場合、重み付けを使い、フィッティング後に縮小カイ二乗値を計算できます。

縮小カイ二乗値は、フィッティング係数の出力された誤差推定値を調整するために使用できます。

この操作を行うと、結果として得られる縮小カイ二乗値は、フィッティングの検定に使用できなくなります。



例えば、X に比例するガウス誤差を持つデータセットを作ります。

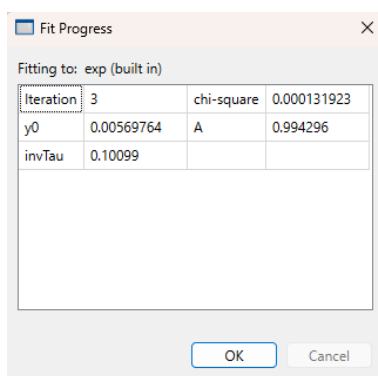
```
Make data = exp(-x/10) + gnoise((x+1)/1000)
Display data
```

X に比例する重みを持つ重み付けウェーブを準備しますが、真の測定誤差とは等しくないものとします。

```
Duplicate data, data_wt
data_wt = x+1 // x に対して正しい比例関係にあるが、1000 倍大きい
```

フィッティングにはほとんど意味のない係数誤差が生じています。
これは与えられた重みが実際の測定誤差に比例していたものの、1000 倍も大きすぎたためです。

```
CurveFit/NTHR=0 exp data /W=data_wt /I=1 /D
```



```
0 •Make data = exp(-x/10) + gnoise((x+1)/1000)
1 •Display data
2 •Duplicate data, data_wt
3 •data_wt = x+1
4 •CurveFit/NTHR=0 exp data /W=data_wt /I=1 /D
5 Fit converged properly
6 fit_data= W_coef[0]+W_coef[1]*exp(-W_coef[2]*x)
7 W_coef={0.0056976,0.9943,0.10099}
8 V_chisq= 0.000131923;V_npnts= 128;V_numNaNs= 0;V_numINfs= 0;
9 V_startRow= 0;V_endRow= 127;
10 W_sigma={5.77,5.73,1.16}
11 Coefficient values ± one standard deviation
12 y0 = 0.0056976 ± 5.77
13 A = 0.9943 ± 5.73
14 invTau = 0.10099 ± 1.16
15 |
```

では、縮小カイ二乗値に基づいて縮小誤差を計算します。

```
Variable reducedChiSquare = V_chisq/(V_npnts - numpnts(W_coef))
Duplicate W_sigma, reducedSigma
reducedSigma = W_sigma*sqrt(reducedChiSquare)
```

結果として生じる誤差はより妥当です。

```
Print reducedSigma
```

```
18 •Print reducedSigma
19 reducedSigma[0] = {0.00592358,0.00588357,0.00118976}
20 |
```

非線形フィッティングと同様に、このようなガウス統計は実際には適用できないことに留意してください。
結果の利用には注意が必要です。

多変量関数へのフィッティング

多変量関数とは、複数の独立変数を持つ関数です。

これは、ある二次元領域上で測定データを取得した場合に生じます。

例えば、様々な地点で表面温度を測定すると、温度が X と Y の両方の関数となる結果が得られます。

また、プロセスの出力が様々な入力（例えば試薬の初期濃度、温度、圧力など）に依存する関係を調べようとする場合にも生じることがあります。

このようなケースでは、独立変数の数が非常に多くなる可能性があります。

多変量関数のフィッティングは、1つの独立変数を持つ関数のフィッティングとほぼ同様です。
ここでの説明は、単変量関数のフィッティングに関するこれまでの説明を既に読んでいることを前提としています。

New Fit Function ボタンをクリックすると、新しい多変量ユーザー定義関数を作成できます。

独立変数リストには、複数の変数名を入力します。
プロシージャウィンドウの行の長さによって設定された十分な制限内であれば、任意の数の独立変数を使用できます。

一変数関数は通常 $y = f(x)$ と表記され、Curve Fitting ダイアログでは入力データを選択するメニューに「Y data」と「X data」というラベルを使うことでこれを反映しています。
多変量データはそれほど便利ではありません。
空間データに適合させる関数はしばしば $z = f(x, y)$ と記述されます。
体積データは $g = f(x, y, z)$ となります。
多数の独立変数を持つ関数はしばしば $y = f(x_1, x_2, \dots)$ と記述されます。

混乱を避けるため、ここではYデータとXデータのラベルをそのまま使い、それぞれ従属変数と独立変数を意味するものとします。

一変量関数と多変量関数の主な違いは、入力データの選択にあります。
独立変数が4つ以下の場合、Y値を保持するために多次元ウェーブを使用できます。
これは空間グリッド上で測定されたデータ、または各独立変数において等間隔で測定されたその他のデータに適しています。
多次元ウェーブ内のデータを「グリッド化データ」と呼びます。

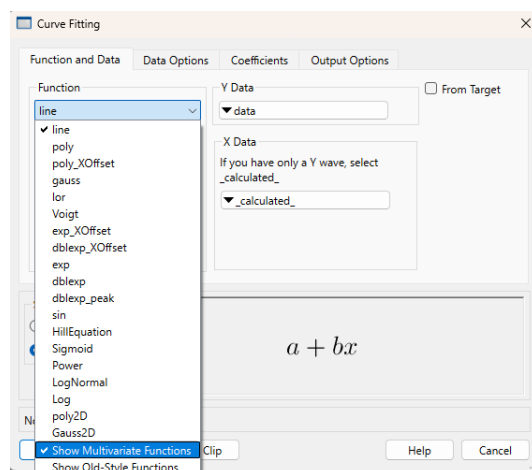
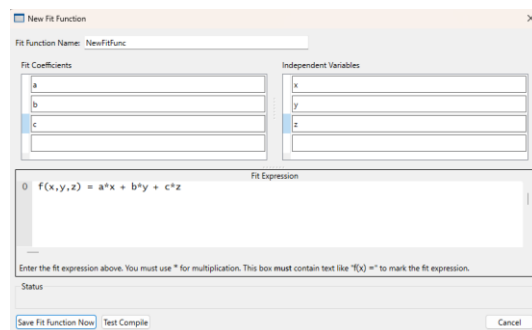
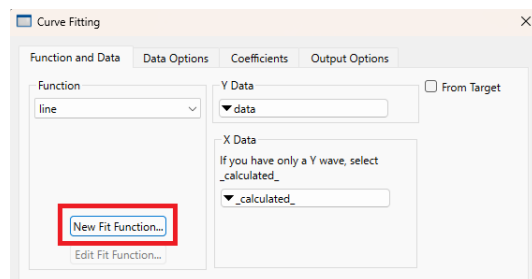
あるいは、Y値を保持するために1Dウェーブを使うこともできます。
独立変数は、各独立変数ごとに1つのウェーブを持つN個の1Dウェーブ、または1つのN列行列ウェーブとして指定できます。
Xウェーブは、Yウェーブと同じ行数を持つ必要があります。

多変量関数の選択

Curve Fitting ダイアログを初めて使う場合、多変量関数はFunctionメニューにリストされません。
まず最初に行うべきことは、多変量関数のリスト表示を有効にすることです。
これを行うには、FunctionメニューのShow Multivariate Functionsにチェックを入れます。
これにより、2つの組み込み多変量関数 poly2D と Gauss2D、および適切なユーザー定義関数がメニューに表示されます。

Show Multivariate Functions 設定は環境設定に保存されます。
再度オフにしない限り、選択する必要はありません。

メニューから多変量関数を選択できるようになりました。



多変量関数に対するフィッティングデータの選択

多変量関数を選択すると、Y Data メニューには 1 次元ウェーブと、フィッティング関数が必要とする独立変数の数に一致する多次元ウェーブが表示されます。

1 次元 Y データウェーブに対する X データの選択

Y データが 1 次元のウェーブである場合、各独立変数に対して X ウェーブを選択する必要があります。

複数の独立変数に対して X スケーリングデータを Y ウェーブに保存する方法はないため、_calculated_ は存在しません。

Y Data メニューで 1 次元ウェーブを選択すると、X Data メニューには 1 次元ウェーブと 2 次元ウェーブの両方が表示され、N 個の独立変数を持つ関数の場合は N 列が表示されます。

X ウェーブを選択すると、ウェーブの名前がメニューの下にあるリストに転送されます。

適切な数のウェーブを選択すると、X Data メニューは無効になります。

X ウェーブを選択する順序は重要です。

最初に選択したウェーブは最初の独立変数の値を提供し、以下同様です。

リストから X データウェーブを削除するには、ウェーブ名をクリックして Backspace キーを押してください。

X データウェーブの順序を変更するには、リストで 1 つ以上のウェーブを選択し、適切な順序にドラッグしてください。

グリッド化された Y データに対する X データの選択

多次元 Y ウェーブを選択する場合、独立変数の値は Y ウェーブの次元スケーリングから取得することも、Y ウェーブの関連次元の値を含む 1 次元ウェーブから取得することもできます。

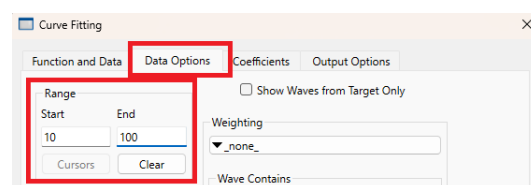
つまり、2 次元マトリックス Y ウェーブがある場合、X 次元の値を提供するウェーブと Y 次元の値を提供するウェーブを選択できます。

Independent Variable メニューには、Y ウェーブの指定された次元に一致するウェーブのみが表示されます。

多変量関数に対するデータの部分範囲のフィッティング

1 次元 Y ウェーブのデータ部分範囲の選択は、単変量関数の部分範囲選択と同様です。

Data Options タブの Start および End 範囲ボックスにポイント番号を入力するだけです。



グリッド化された Y データをフィッティングする場合、Data Options タブには多次元ウェーブの各次元の開始範囲と終了範囲を設定する 8 つのボックスが表示されます。

これらのボックスに行番号、列番号、レイヤー番号、またはチャンク番号を入力してください。

Y ウェーブがグラフ上で画像として表示される行列ウェーブの場合、カーソルを使ってデータのサブセットを選択できます。

グラフをターゲットウィンドウとして、カーソルボタンをクリックすると、範囲ボックスにテキストが入力され、この操作が行われます。

コンタープロットでのカーソル機能の使用は単純ではなく、ダイアログでは対応していません。

データマスクウェーブを使ってデータの部分範囲を選択することもできます（「マスクウェーブを使う」のセクションを参照）。

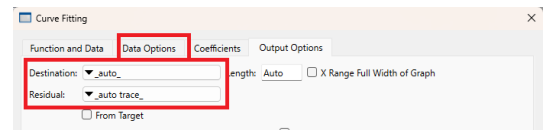
データマスクウェーブは、Y データウェーブと同じポイント数と次元を持つ必要があります。

多変量フィッティングのモデルの結果

1 つの独立変数の関数へのフィッティングと同様に、モデル出力と残差を含むウェーブを自動的に作成します。

これは、Output Options タブで、保存先として `_auto_`、残差として `_auto trace_` を選択すると実行されます。

ただし、詳細にはいくつかの違いがあります。



デフォルトでは、単変量フィッティングのモデル曲線は、モデルフィッティングを表示するために 200 ポイントの滑らかな曲線です。

これは、独立変数の連続する値の間を合理的に補間できることに依存しています。

一方、複数列の独立変数の場合、すべての独立変数の連続した値が順番に並んでいるとは限らないため、この方法は適用できません。

その結果、従属変数データウェーブの各ポイントに対してモデルポイントを計算します。

データウェーブが最前面のグラフウィンドウに単純な 1D トレースとして表示されている場合、フィッティング結果がグラフに追加されます。

残差は常にポイントごとの計算に基づいて算出されるため、多変量多列フィッティングの残差計算は単変量フィッティングと同じです。

多次元ウェーブへのフィッティング結果の表示はより問題が多くなります。

従属変数が 3 次元以上の場合、結果の表示は容易ではありません。

モデルウェーブと残差ウェーブは生成され、結果は計算されますが、表示はされません。

Gizmo を使って様々な 3D プロットを作成できます。

Windows→New→3D Plot メニューから適切なプロットタイプを選択するだけです。

Y データウェーブが画像またはコンターとして最前面のグラフに表示されている場合、2 次元行列ウェーブへのフィッティング結果が最前面のグラフに表示されます。

データが画像またはコンターとして表示されているかに関わらず、モデル結果はコンターとしてプロットされます。画像として表示されたデータの上にコンターでプロットされたモデル結果は、非常に強力な可視化手法です。

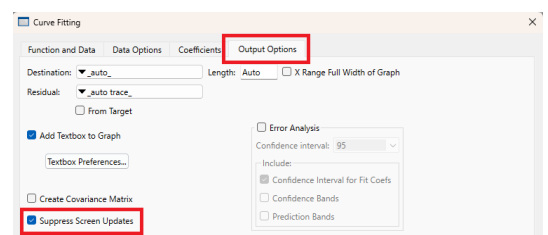
残差は、データと同様の方法で、自動的に作成される別のグラフウィンドウに表示されます。

ウィンドウサイズは、データを表示するウィンドウと同じになります。

表示の更新に必要な時間

コンターや画像の再描画にはかなりの時間がかかるため、反復処理ごとの表示更新により、コンターや画像データの描画が非常に遅くなる場合があります。

更新を抑制するには、Output Options タブの Suppress Screen Updates チェックボックスをクリックしてください。



多変量フィッティングの例

多変量関数へのフィッティングの例を2つ示します。

最初の例では、組み込み関数 `poly2D` を用いて格子データセットに平面をフィッティングさせ、データから平面的な傾向を除去します。

2つ目の例では、簡略化された2次元ガウス関数を定義し、ランダムなXYZデータを使ってXY空間におけるピークの位置を定義します。

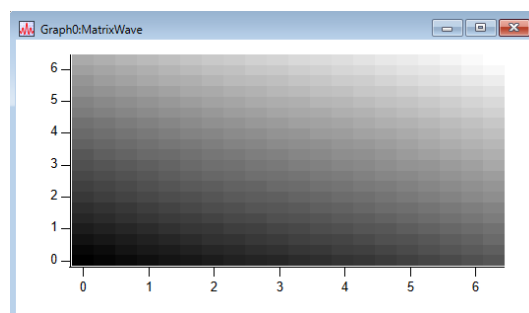
例1 - Poly2D を使って平面トレンドを除去する

行列に二次元正弦ウェーブを埋め込み、平面トレンドが正弦ウェーブを圧倒する例を示します。

この例では、トレンドを除去するためにデータに平面をフィットさせる方法を示します。

まず、データ行列を作成し、値を埋め込み、行列を画像として表示します。

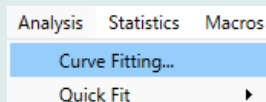
```
Make/O/N=(20,20) MatrixWave  
SetScale/I x 0,2*pi,MatrixWave  
SetScale/I y 0,2*pi,MatrixWave  
MatrixWave = sin(x) + sin(y) + 5*x + 10*y  
Display;AppendImage MatrixWave
```



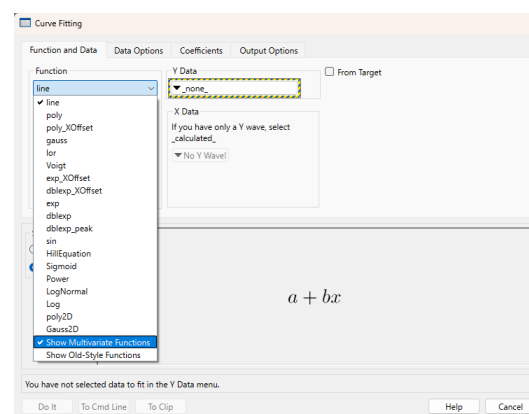
このコマンドは、右のような画像を表示するグラフを作成します。
左下から右上へのグラデーションに注目してください。

これでフィッティングの準備が整いました。

1. Analysis メニューから Curve Fitting を選択すると、Curve Fitting ダイアログが表示されます。

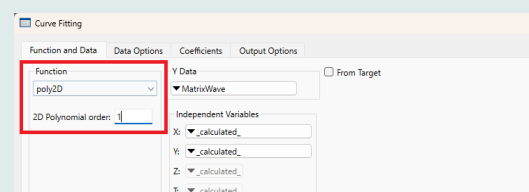


2. まだ行っていない場合は、Function メニューから Show Multivariate Functions を選択してください。

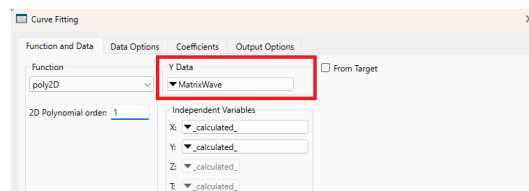


3. Function メニューから Poly2D を選択します。

2D Polynomial Order を 1 に設定していることを確認してください。

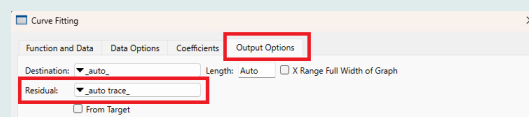


4. Y Data メニューから MatrixWave を選択してください。



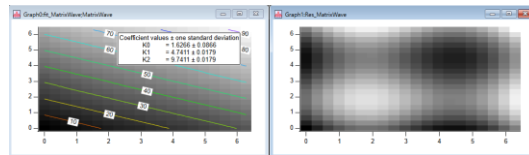
5. Output Option タブをクリックしてください。

Residual メニューから _auto trace_ を選択してください。



6. Do It をクリックします。

結果として、元のグラフにデータへのフィッティングを示すコンター
プロットと、フィッティング後に残った正弦波信号を示す残差の
新しいグラフが表示されます。



同様に、ImageRemoveBackground コマンドを使用できます。

これは同じ処理をワンステップで行うコマンドです。

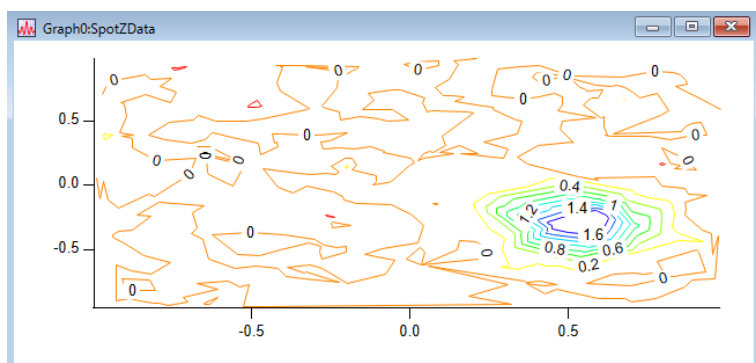
画像プロットを最前面のウィンドウに表示している場合、Image メニューに Remove Background が表示されま
す。

例 2 - ユーザー定義の簡易 2 次元ガウスフィッティング

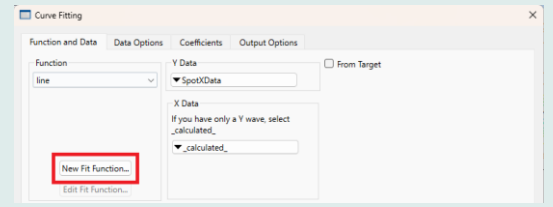
この例では、スポットの中心を見つけるために、2次元ガウス関数でフィッティングさせたいスポットを定義するデ
ータがあります。

何らかの理由で、このデータはランダムな X 座標と Y 座標を持つ XYZ トリプレットの形式で与えられています。
次のコマンドでサンプルデータを生成します。

```
Make/N=300 SpotXData, SpotYData, SpotZData
SetRandomSeed 0.5
SpotXData = enoise(1)
SpotYData = enoise(1)
// {0.55, -0.3}を中心とするガウス分布を作成
SpotZData = 2*exp(-((SpotXData-.55)/.2)^2 - ((SpotYData+.3)/.2)^2)+noise(.1)
Display; AppendXYZContour SpotZData vs {SpotXData,SpotYData}
```

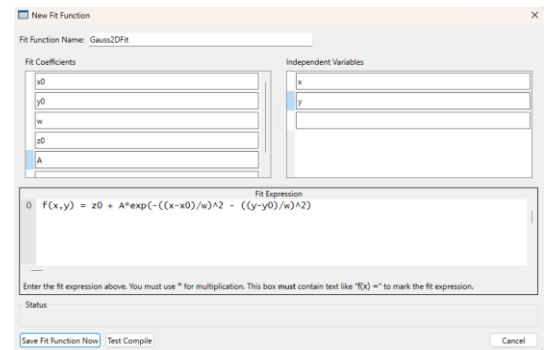


1. Curve Fitting ダイアログを開き、New Fit Function ボタンをクリックして、ユーザー定義のフィッティング関数を入力できるようにします。

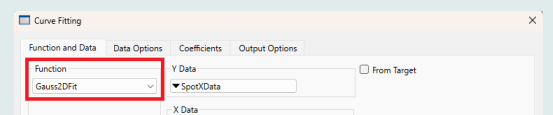


2. スポットが円形であると推測されるため、ガウス関数はX方向とY方向で同じ幅を使用でき、交差相関項は不要です。したがって、新しい関数には、ベースラインオフセットの z0 係数、振幅の A 係数、X 座標と Y 座標の x0 および y0 係数、幅の w 係数が含まれます。

右は New Fit Function ダイアログでの入力例です。

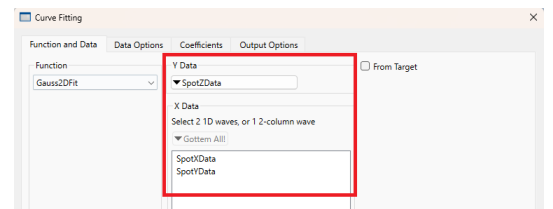


3. Save Fit Function Now をクリックすると、関数が手順ウィンドウに保存され、Curve Fitting ダイアログに戻ります。新しい関数は自動的に Function メニューで選択されます。

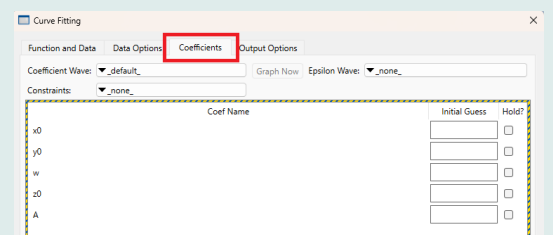


フィッティングを実行します。

**4. Y Data メニューから SpotZData を選択します。
X Data メニューから SpotXData を選択します。
X Data メニューから SpotYData を選択します。**

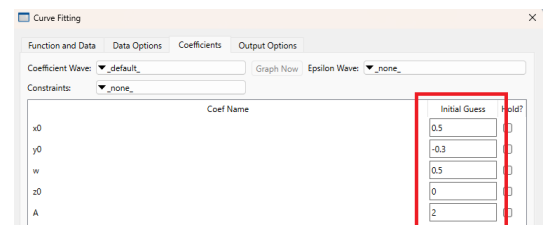


5. Coefficients タブをクリックします（下のエラーボックスは初期推定値を入力する必要があることを示しています）。



6. 初期推定値を入力します。

z0 = 0
A = 2
x0 = 0.5
y0 = -0.3
w = 0.5



この問題では、残差と宛先は重要ではありません。
単にスポット中心の座標を知りたいだけだからです。

7. Do It をクリックすると、履歴に次が表示されます。

```
9 •FuncFit/TBOX=768 Gauss2DFit W_coef SpotZData /X={SpotXData,SpotYData} /D
10 Fit converged properly
11 fit_SpotZData= Gauss2DFit(W_coef,x,y)
12 W_coef={0.54091,-0.30026,0.20355,-0.011116,2.0323}
13 V_chisq= 2.72757;V_npnts= 300;V_numNaNs= 0;V_numINFs= 0;
14 V_startRow= 0;V_endRow= 299;
15 W_sigma={0.00512,0.0053,0.00537,0.00591,0.0566}
16 Coefficient values ± one standard deviation
17 x0 = 0.54091 ± 0.00512
18 y0 = -0.30026 ± 0.0053
19 w = 0.20355 ± 0.00537
20 z0 = -0.011116 ± 0.00591
21 A = 2.0323 ± 0.0566
22 |
```

Fit Progress			
Fitting to: Gauss2DFit			
Iteration	5	chi-square	2.72757
x0	0.540915	y0	-0.300261
w	0.203551	z0	-0.0111162
A	2.03231		

出力結果から、フィッティングによりスポットの中心が
{0.54091, -0.30026} と決定されたことが示されています。

Curve Fitting ダイアログの問題

Curve Fitting ダイアログを使う時、期待通りに動作しない場合があります。
よくある問題としては：

Function メニューでユーザー定義関数が見つからない。

これは通常、次の2つの理由のいずれかによって発生します。
関数が多変数関数であるか、古い形式の関数であるかのいずれか
です。

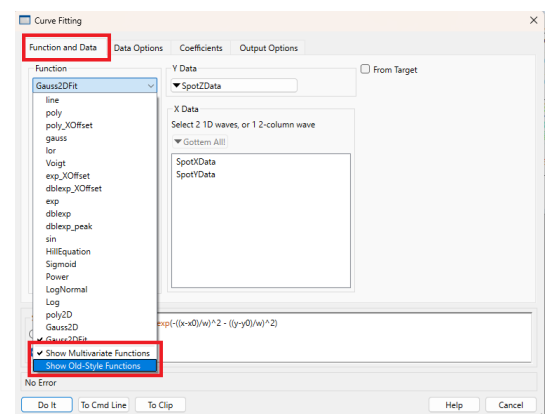
この問題は、Function and Data タブの Function メニューから
Show Multivariate Functions または Show Old-Style
Functions を選択することで解決されます。

Show Old-Style Functions を選択するとフィッティング関数が
表示される場合は、Edit Fit Function ボタンをクリックしま
す。

これにより、Edit Fit Function ダイアログが表示されます。

ダイアログの初期化の一部として、現在の標準に準拠するようにフィッティング関数が修正されます。

このダイアログで、フィッティング係数に覚えやすい名前を付けることもできます。



「Igor can't determine the number of coefficients (Igor は係数の数を決定できません)」というメッセージが表示される。

外部関数またはユーザー定義関数を使っている時に、その関数が複雑すぎてダイアログが関数コードを解析して必要
な係数の数を判断できない場合、Coefficients タブをクリックするとこの現象が発生します。

この問題を回避する唯一の方法は、明示的な係数ウェーブ（「係数ウェーブ」のセクションを参照）を選択すること
です。

ダイアログは係数ウェーブのポイント数に基づいて係数の数を決定します。