# 必用なもの

python version 3

SciPy version 0.17.0以上

Matplotlib バージョンへの制限は不明

NumPy バージョンへの制限は不明

# 使用方法

usage: fit.py [-h] [-without\_plot] [-without\_fit] [-sample\_range SAMPLE\_RANGE]

[-method {lm,trf,dogbox}] [-en EN]

fnames area dbfname outfname

positional arguments:

fnames Input file names

area Area

dbfname database file name for initial parameters

outfname output file name

optional arguments:

-h, --help show this help message and exit

-without\_plot Without result plot

-without\_fit Without fit

-sample\_range SAMPLE\_RANGE

Sample wavelength ranges

-method {lm,trf,dogbox}

Algorithm to perform minimization. Default:trf

-en EN Extension number. Default:0

**fname**

データキューブFITSファイル

**Area**

フィッティングに用いる領域

コンマで区切って次のように記載する。　X1,X2,Y1,Y2

**dbfname**

入力のデータベースファイル。

１行目　ファイルの作成日時

２行目　データキューブFITSファイル名

３行目　フィットに用いた領域

４行目　フィットに用いた波長範囲

５行目　入力データファイル名

６行目　フィットに用いたアルゴリズム名

７行目　フィットに用いたinput database file名とフィット終了時のメッセージ

８行目以降　フィットに用いる関数のパラメーター情報

最初の行は関数名と、その関数が利用するパラメーター数が記載されている。

次からパラメーターの情報で、最初の数値からそれぞれ以下の情報を表している。

初期値、下限値、上限値、ステップ、収束条件、フラグ

上下限値に制限を設けない時には十分広い範囲を指定する。

ステップはフィッティングの際に最初に用いられるステップ。フラグでリンクを選択した場合には、リンク先に対するスケーリングファクター。

フラグは固定値にしたければ-１、最適化したければ0、他のパラメーターとリンクさせたければ、その成分番号。

**outfname**

フィット結果が記載されたデータベースファイル

dbfnameと同じフォーマットにしているので、出力結果を入力データベースとして使える。

**-without\_plot**

フィット後に結果をプロット表示しない。

**-without\_fit**

フィットせずに初期パラメーターを用いて表示だけ行う。初期パラメーターのチェック用。

これを指定した場合、アウトプットファイルは本来必要ないが、指定しておかないとエラーとなる。

**-sapmple\_range SAMPLE\_RANGE**

フィッティングに用いる波長範囲を指定する。波長範囲（SAMPLE\_RANGE）の記述方法は 6500-6600,7000-7300 のように複数の範囲をコンマでつなげて指定することもできる。＊（アスタリスク）を指定すると、全ての範囲を用いる（特に指定しなくてもデフォルト設定になっている）。

**-method {lm,trf,dogbox}**

フィットのアルゴリズム。デフォルトはtrf。

**-en EN**

データキューブFITSファイルにおいて、フィットに用いるデータが格納されているextension番号。デフォルトは0。

# 関数とパラメーター

## gaussian

パラメーター数：３

パラメーター１：ピーク強度

パラメーター２：中心波長　（単位はデータファイルで使用されている単位）

パラメーター３：輝線幅（km/s）

## lorentzian

パラメーター数：３

パラメーター１：ピーク強度

パラメーター２：中心波長　（単位はデータファイルで使用されている単位）

パラメーター３：輝線幅（km/s）

## linear

パラメーター数：２

パラメーター１：傾き

パラメーター２：波長０での強度

## 関数の追加

funcs.pyに関数情報を追加すればよい。

関数自体の定義と、パラメーターのスケーリング方法を定義する。