Travaux Pratiques des cours

Traitement des données expérimentales de la physique

et

Analyse de séries temporelles

Guy Munhoven, Sophie Pedoux, Emmanuel Jehin et Pierre Magain

Version des encadrants (janvier 2010)

Introduction

Premiers pas dans MIDAS

Petite initiation à MIDAS

Après avoir ouvert une session sous LINUX sur votre station de travail, ouvrez une fenêtre terminal (`xterm`, `konsole`). Passez dans le répertoire de travail `TP-Midas`, sous le répertoire de login à l’aide de la commande

```
cd TP-Midas
```

entrée à partir du prompt de la fenêtre terminal.

Pour lancer l’application MIDAS, taper `inmidas` à partir de ce même prompt. Une fois MIDAS lancé, il est conseillé d’ouvrir tout de suite une fenêtre graphique (normalement, il y en a une qui s’ouvre automatiquement au démarrage) à l’aide de la commande

`create/graphics`.

Beaucoup de commandes MIDAS ont cette structure `commande/qualificatif`. Le `qualificatif` optionnel peut généralement être abrégé par ses trois premières lettres.

Vous pouvez librement déplacer les deux fenêtres à l’écran. **Evitez cependant absolument d’en modifier les tailles.** Le fonctionnement de MIDAS s’en trouverait fortement perturbé.

Données

`d32` (32 points).

Description

Ensemble de points décrivant une droite de pente 1, allant de -16 à $+15$, par pas de 1.

Tâches

Visualisation :

`plot d32`	afficher sous forme de courbe
`plot/pt d32`	afficher les points de données isolés
`plot/pl d32`	afficher les points de données reliés

Pour superposer en un deuxième temps un autre jeu de données stocké dans le fichier *nom_de_fichier*, utiliser

`overplot` *nom_de_fichier*

On peut aussi, à l'aide des opérations algébriques courantes, créer d'autres fichiers :

`compute nouvar␣=␣`*opérations mathématiques*

A noter qu'il faut impérativement saisir un espace (␣) des deux côtés du signe égal (=). Tous les jeux de données que vous créez de cette manière sont stockés sous forme de fichier sur le disque dur de votre poste de travail. Ils ne seront pas perdus en sortant de votre session MIDAS et vous pourrez les réutiliser ultérieurement. Le nom du fichier sous lequel on désire retrouver les nouvelles données (ici `nouvar`), peut être librement choisi. Il est recommandé de limiter les noms à six lettres, et de ne pas compter sur une distinction entre majuscules et minuscules. Des noms plus longs fonctionnent parfois, parfois pas; la distinction minuscule/majuscule fonctionne parfois, parfois pas.

Exemple

```
compute testsi = 10.*sin(20.*d32) + 4.
plot d32
overplot testsi
```

Le format utilisé par MIDAS pour les jeux de données que nous utiliserons, tout comme celles que vous allez créer par vous-mêmes, ont un certain nombre d'informations associées, telles que

- le nombre de dimensions (toujours 1 dans le cadre de ces travaux pratiques);
- le nombre de points de données;
- l'extension des axes (i.e., du seul axe des x dans notre cas).

La commande MIDAS donnant accès à cette information est

`read/descriptor` *nom_de_fichier*

Cette commande affichera toute une série d'informations, dont le nombre de points (`npix`) est ce qui nous intéresse plus particulièrement dans un premier temps.

Exercice 1

Données

`d128` (...... points) ou `d256` (...... points)

Description

Droite allant de -64 à $+63$ (resp. de -128 à $+127$) par pas de 1.

Tâches

Calcul des transformées de Fourier de Sinc et Sinc^2, c'est-à-dire, ici de

$$\mathrm{Sinc}(x) = \frac{\sin(360 * f * x)}{2\pi * f * x}$$

et de son carré. Nous choisirons $f = 0.1$ comme fréquence. Noter cette définition inhabituelle de $\mathrm{Sinc}(x)$, due au fait que MIDAS attend l'argument du sinus en degrés, alors que l'argument du sinus en mathématiques est en radians (Attention : si on exprime les arguments des fonctions trigonométriques en degrés—ce qui est pourtant tout à fait légitime—même les formules les plus élémentaires de dérivation ne sont plus applicables telles quelles.)

Etapes

Sinc n'existe pas d'office sous MIDAS, et il faut donc d'abord le créer

```
compute x = d256
compute sinc = sin(0.1*360*x)/(0.1*2.*3.1416*x)
```

MIDAS signale une erreur. Que signifie-t-elle ? Inspection :

```
plot/pl sinc
```

Pour y remédier, il faut manuellement corriger `sinc`:

```
write sinc 0.,1 1.
```

Ré-inspection :

```
plot/pl sinc
```

Pour effectuer la transformée de Fourier, il faut encore compléter les données de départ de leur partie complexe, même si celle-ci est identiquement nulle. Elle doit absolument avoir les mêmes caractéristiques (nombre de points, pas d'échantillonnage) que la partie réelle. Il existe plusieurs moyens de le faire:

```
compute zero = sinc-sinc
compute zero = sinc*0
compute zero = x-x
compute zero = x*0
```

Noter que `sinc` et `x` ont les mêmes caractéristiques (i.e., dimensions, et espacement des points). Puis calculer la transformée de Fourier directe à l'aide de la fonction `tfd`

```
tfd sinc zero rfsinc ifsinc
```

qui retourne les parties réelle et imaginaire dans les fichiers (variables) en troisième (`rfsinc`) et quatrième (`ifsinc`) arguments, respectivement. Les noms de ces variables peuvent être librement choisis.

Inspecter et commenter les résultats (parties réelle et imaginaire).

Sinc^2 Le carré de Sinc peut immédiatement être calculé de ce qui précède :

```
compute sinc2 = sinc*sinc
```

ou

```
compute sinc2 = sinc**2
```

en utilisant l'opérateur d'exponentiation `**`. Calculer le transformée de Fourier de `sinc2` (quelle partie imaginaire utiliser ?). Vérifier les résultats et commenter.

Exercice 2

Données

`d128` (...... points)

Description

Droite de pente unité, allant de -64 à $+63$, par pas de 1.

Tâches

Calcul la transformée de Fourier de la Gaussienne

$$\mathrm{G}(x) = a \exp(-b * (x-c)^2)$$

qui a pour amplitude $a = 12.0$ et pour centre $c = 3.4$; b sera déterminé de manière à ce que la largeur à mi-hauteur de la Gaussienne soit égale à 6, et x balayera l'intervalle $[-64, 63]$ ($x =$ `d128`).

Etapes

1. Déterminer b.

2. Calculer la Gaussienne et sa transformée de Fourier

   ```
   compute gauss = 12.*exp(-0.077*(d128-3.4)**2)
   plot gauss
   compute zero = d128*0
   tfd gauss zero rg ig
   plot rg
   ```

   Observations ? Commentaires ? Explications ?

Fin de séance

Pour sortir de la session MIDAS, taper

```
bye
```

Pour fermer la session LINUX/KDE, sélectionnez "Log Out" sous le menu marqué par le grand "K". Pour arrêter proprement votre station de travail, utilisez, s'il-vous-plaît, le bouton "Turn off Computer" dans le menu de dialogue qui suit, ou "Restart Computer" si vous désirez redémarrer (sous LINUX ou sous WINDOWS).

Interpolation par transformée de Fourier

Exercice 1

Données

`data1`

Tâche

Interpoler `data1` avec un pas d’échantillonnage de $1/4$.

Méthodologie

Par transformée de Fourier, avec extension par des zéros.

Etapes

Utilisez `read/descriptor` por déterminer le nombre de points qu’il y a dans le jeu de données. Calculez la transformée de Fourier de `data1` et étendez-la par des zéros pour en arriver au nombre de points requis à l’aide de la fonction `expand`.

Exercice 2

Données

`data2`.

Description

`data2` = `data1` + bruit.

Tâche

Interpoler `data2` avec un pas d’échantillonnage de $\frac{1}{4}$.

Méthodologie

Par transformée de Fourier, avec extension par des zéros.

Exercice 3

Données

`data3`.

Tâche

Interpoler `data3` avec un pas d'échantillonnage de $\frac{1}{8}$.

Méthodologie

Par transformée de Fourier, avec extension par des zéros.

Corrélations, décalages et filtrages

Exercice 1

Données

`t1n` et `t2n`.

Description

Enregistrement d'un même signal, à un décalage près, et avec des erreurs de mesure très importantes.

Tâche

Déterminer le décalage temporel par corrélation croisée.

Méthodologie

Se servir des propriétés de l'opérateur de corrélation sous transformée de Fourier:

$$\mathscr{F}(\mathrm{Corr}(f,g)) = (\mathscr{F}f)^* \times \mathscr{F}g,$$

ce qui donne

$$\mathrm{Corr}(f,g)) = \mathscr{F}^{-1}((\mathscr{F}f)^* \times \mathscr{F}g).$$

Le résultat brut de ces deux opérations sera très bruité, et il faudra recourir à un filtrage.

Etapes

1. Calculer les transformées de Fourier de `t1n` et `t2n`, puis la fonction de corrélation.
2. Programmer un filtre, à générer par l'utilitaire FORTRAN `genfunc.f`:
   (a) ouvrir `genfunc.f` dans un éditeur de texte (comme, par exemple, `vi`, `textedit`, `kedit`, `kwrite`, ...)
   (b) définir les valeurs des paramètres du filtre

   `naxis =` *nombre de dimensions*
   `npix(1) =` *nombre de points de la première dimension*
   `start(1) =` *abscisse de départ de la première dimension*
   `step(1) =` *pas d'échantillonnage de la première dimension*

Ici, le nombre de dimensions est 1, le nombre de points de la première dimension est 256 et l'indice de départ est -0.5. Le pas d'échantillonnage est 0.390625E$-$02. Ces informations peuvent être obtenues en appliquant la commande `read/descriptor` de MIDAS à l'une des deux composantes de la transformée de Fourier déjà calculée (`rfc` ou `ifc` ci-dessus).

Puis inclure la définition du filtre (rectangle, triangle, ...) plus bas, après `subroutine func(......)`, en se servant de la boucle

```
   do 2 i = ...
   y(i) = ...
 2 continue
```

3. Générer le filtre (compiler `genfunc.f` et générer le fichier qu'il définit) et l'appliquer:

```
fortran genfunc
genfunc filtre
compute ... = ... * filtre
```

Exercice 2

Données

`t3n` et `t4n`

Description

Tâche

Déterminer le décalage.

Méthodologie

Par corrélation croisée, via transformée de Fourier et filtrage.

Etapes

Comme pour l'exercice précédent.

Fenêtrage et découpage

Exercice 1

Données

`d128` (128 points)

Description

Droite de pente unité, allant de -64 à $+63$, par pas de 1.

Tâches

Calcul de transformées de Fourier de $\cos(90° * \texttt{d128})$ et de $\cos(80° * \texttt{d128})$.

Exercice 2

Données

`spec2`.

Description

Signal périodique.

Tâche

Déterminer la période.

Méthodologie

`spec2` étant périodique, la transformée de Fourier devrait se présenter sous forme de deux pics δ. Cependant, `spec2` ne présente pas un nombre entier de périodes. Elle est donc égale à une fonction périodique infinie multipliée par un rectangle. En conséquence, sa transformée de Fourier est égale à une somme de deux pics δ, convoluée avec un sinus cardinal (sinc). Il faut donc recourir à un fenêtrage des données. Il convient d'utiliser un triangle à cette fin, car un rectangle serait trop brutal, et générerait nombre d'oscillations de Gibbs.

Etapes

Générer la fenêtre à l'aide de `genfunc`, en prenant soin de travailler dans l'espace des données cette fois-ci, et non pas dans l'espace de Fourier, comme pour les filtres.

Exercice 3

Données

`spec1`.

Description

Signal périodique à plusieurs fréquences.

Tâche

Déterminer le nombre de fréquences et leurs valeurs.

Méthodologie

Par transformée de Fourier.

Analyse spectrale par transformée de Fourier

Exercice 1

Données

`spec5`.

Description

Signal à deux composantes.

Tâche

Effectuer une analyse spectrale du signal: déterminer la fréquence du signal.

Méthodologie

Par transformée de Fourier.

Exercice 2

Données

`spec4`.

Description

Signal composé d'une oscillation à amplitude importante, greffée sur une tendance.

Tâche

Isoler la tendance à long-terme.

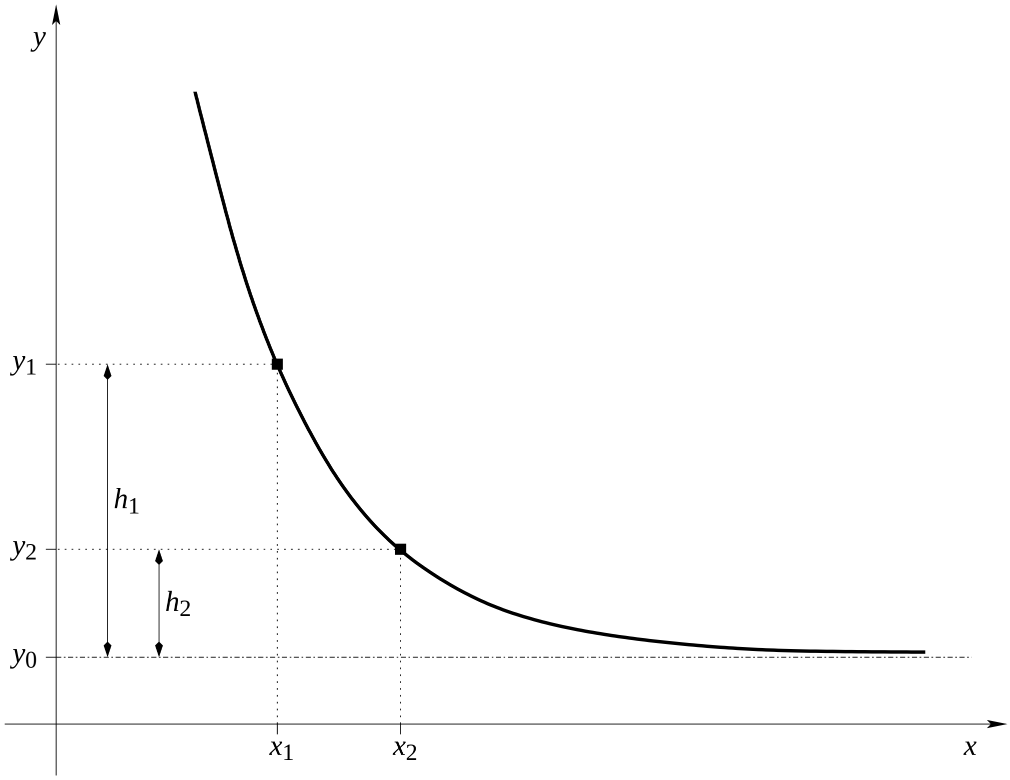
Méthodologie

Par transformée de Fourier, avec filtrage.

Ajustements de modèles

Modèles: exercices préliminaires

Modèle exponentiel décroissant

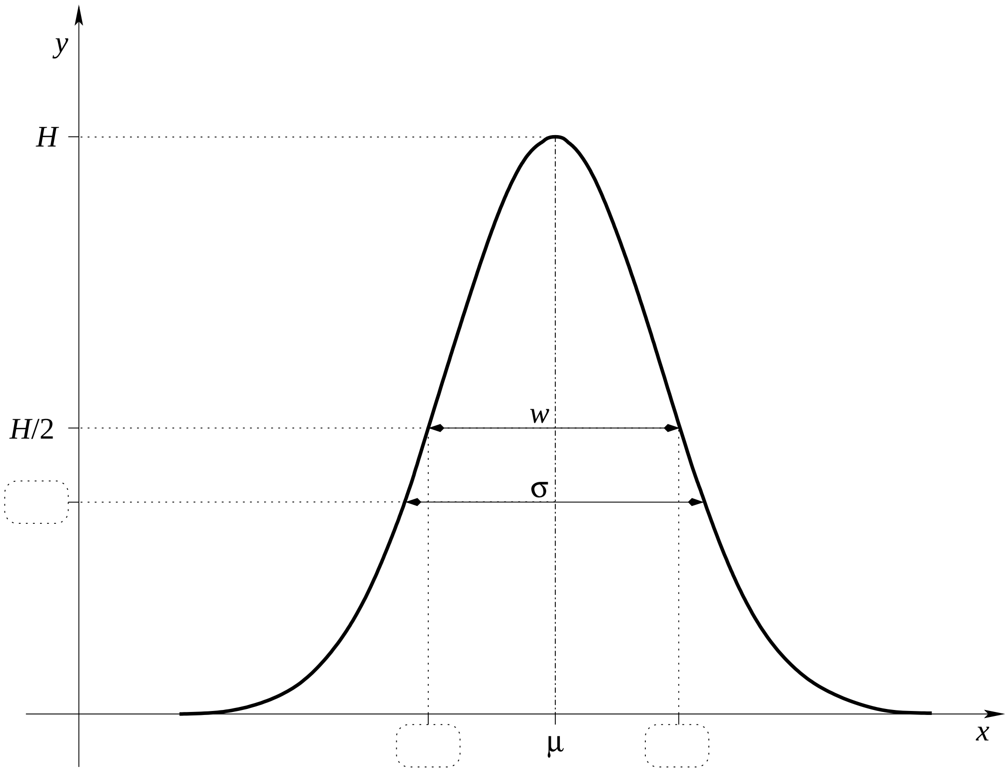


Relier les paramètres a_1, a_2 et a_3 du modèle décrit par l'expression

$$y = a_1 + a_2 \exp(-a_3 x)$$

aux différentes grandeurs indiquées sur la figure ci-dessus.

Modèle exponentiel décroissant



Relier les paramètres a_1, a_2 et a_3 du modèle gaussien décrit soit par par l'expression

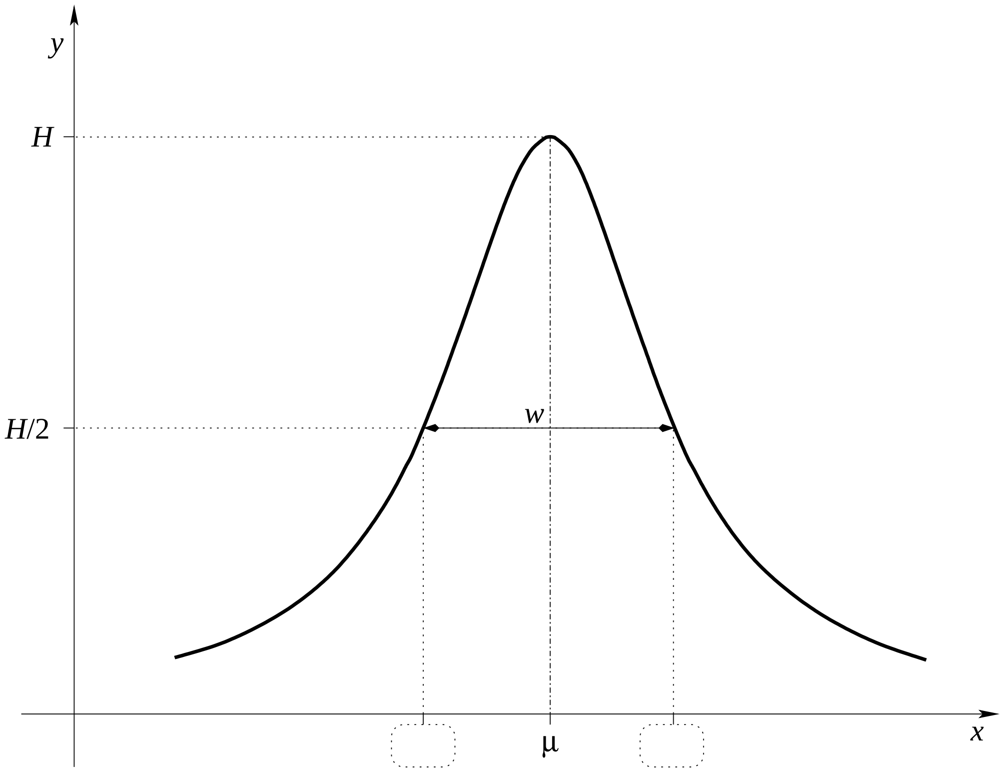
$$y = a_1 \exp\left(-a_2 (x - a_3)^2\right)$$

soit par l'expression

$$y = a_1 \exp\left(-\left(\frac{x - a_3}{a_2}\right)^2\right)$$

aux différentes grandeurs indiquées sur la figure ci-dessus. w est la *largeur à mi-hauteur* de la Gaussienne, μ sa moyenne.

Modèle exponentiel décroissant



Relier les paramètres a_1, a_2 et a_3 du modèle lorentzien décrit soit par par l'expression

$$y = \frac{a_1}{a_2 + (x - a_3)^2}$$

soit par l'expression

$$y = \frac{a_1}{1 + a_2(x - a_3)^2}$$

aux différentes grandeurs indiquées sur la figure ci-dessus. w est la *largeur à mi-hauteur* de la Lorentzienne, μ sa moyenne.

Modèle 1: décroissance radioactive

Données

`spec7`.

Description

Enregistrement d'un comptage de particules au cours d'une expérience de décroissance radioactive.

Tâche

Déterminer les caractéristiques (nombre initial, temps caractéristique).

Méthodologie

Par ajustement de modèle.

Etapes

1. Editer `ajuste.f`:
   (a) adapter partout le nombre de paramètres (5 lignes concernées)
   `parameter (ma=`*nombre_de_paramètres*`)`;
   (b) dans `subroutine funce(...)`, programmer la fonction qui décrit le modèle
   `y =` *fct*(`x`, `a(1)`, ..., `a(ma)`),
   ainsi que toutes ses dérivées partielles par rapport aux paramètres:
   `dyda(1) =` $\partial y/\partial a_1$(`x`, `a(1)`, ..., `a(ma)`),
   ...,
   `dyda(ma) =` $\partial y/\partial a_{\mathrm{ma}}$(`x`, `a(1)`, ..., `a(ma)`).
2. Compiler `ajuste.f` :

   ```
   fortran ajuste
   ```

3. Déterminer les σ pour chaque point de mesure.

   ```
   compute sigma = ...
   ```

4. Déterminer des estimations initiales pour les paramètres :

- $a(1) \simeq \ldots$
- $a(2) \simeq \ldots$

5. Effectuer l’ajustement :

   `ajuste spec7 sigma resfit` a_1`,`a_2

6. `ajuste` procède à une minimisation du χ^2 et donne:

   - les caractéristiques des itérations intermédiaires;
   - matrice des covariances, qui fournit les informations sur les barres d’erreur des paramètres: ainsi $cov(i,i) = \sigma_i^2$; $cov(i,j)$ caractérise la corrélation entre a_i et a_j;
   - le χ^2 final;
   - le Q, qui, *grosso modo*, donne la probabilité qu’un χ^2 élevé serait dû au hasard.

Modèle 2: décroissance radioactive

Données

`spec9`

Description

Enregistrement d’un processus de décroissance radioactive.

Tâche

Déterminer les caractéristiques de la source radioactive.

Méthodologie

Par ajustement de modèle.

Etapes

1. Ajustement d’une exponentielle décroissante.

   L’ajustement est-il acceptable ? Justification ? Si non, comment améliorer l’ajustement ?

Modèle 3: isoler une perturbation

Données

`spec6`.

Description

Sinus + perturbation.

Tâche

S'affranchir du sinus porteur et isoler la perturbation.

Méthodologie

Par ajustement de modèle (avec `ajuste.f`) et non pas par filtrage; en filtrant le sinus dominant, on risque de ne plus retrouver la perturbation, ou alors de sérieusement la déformer.

Il est donc préférable de procéder de la manière suivante: détecter une partie non perturbée du signal, y effectuer un ajustement de modèle en sinus (ou cosinus), et puis soustraire le modèle de la série brute afin de faire apparaître la perturbation. La perturbation est très faible, il faut donc procéder par étapes. Le fit de sinus sur plusieurs périodes risque de poser problème à la routine de minimisation utilisée pour le fit, et il convient d'avoir une bonne estimation de départ pour les paramètres. Utiliser un sigma de 0.01 partout.

Etapes

1. Extraire une période à peu près et effectuer un ajustement dessus.

2. Etendre ce modèle à toute l'étendue des x; le soustraire du signal pour avoir une meilleure idée de l'étendue de la perturbation.

3. Extraire la plus grand partie non perturbée possible, et répéter l'ajustement précédent pour améliorer les valeurs des paramètres, en utilisant les valeurs de paramètres ajustés à la première étape comme valeurs initiales.

Modèle 4: instrument imparfait

Données

`spec8`

Description

Enregistrement de trois raies en émission, effectué à l’aide d’un instrument caractérisé par un bruit gaussien identique en chaque pixel (à déterminer).

Tâche

Déterminer les caractéristiques des raies (position, intensité, largeur) ainsi que du bruit gaussien.

Méthodologie

Par ajustement de modèle (`ajuste.f`), à l’aide d’un modèle à déterminer au mieux.

Modèle 5: instrument décalibré

Données

`spec10`

Description

Enregistrement d’un comptage de particules avec un instrument caractérisé par

- un point zéro indéterminé;
- un bruit gaussien.

Le pic est Lorentzien.

Tâche

Déterminer la constante de temps de la décroissance radioactive ainsi que les caractéristiques de la Lorentzienne, ainsi que les paramètres de l’instrument. Question subsidiaire: estimer la barre d’erreur sur l’amplitude de la Lorentzienne.

Méthodologie

On procèdera à un ajustement de modèle de manière itérative, étant donné que certaines caractéristiques (e.g. σ) ne sont pas connues dès le départ, et qu’il faut donc commencer par quelques approximations plus ou moins grossières.

Modèle 6: filtre de Wiener

Données

`spec11`.

Description

Spectre enregistré avec un instrument qui a pour profil instrumental `profil11` — utiliser `profil11b` pour le traitement.

Tâche

Déconvoluer le signal enregistré à l'aide d'un filtre de Wiener, afin de pouvoir juger si le signal réel présente une raie large, ou plusieurs raies étroites isolées voisines (combien ?).

Méthodologie

Par construction du filtre de Wiener (voir chapitre 9 du cours pour la procédure à suivre).

Exercices de révision

Exercice 1

Données

`t5n` et `t6n`.

Tâche et méthodologie

Déterminer le décalage par corrélation croisée, via transformée de Fourier et filtrage éventuel.

Etapes

Comme pour l'exercice de la séance 3.

Exercice 2

Données

`spec14` et `spec15`.

Tâche

Interpôler `spec14` et `spec15` avec un pas d'échantillonnage de $1/4$. Discuter la qualité de l'interpolation.

Exercice 3

Données

`spec17`.

Description

Signal composé d'une oscillation à amplitude importante, greffée sur une tendance.

Tâche et méthodologie

Isoler la tendance à long-terme, par transformée de Fourier, avec filtrage.

Exercice 4

Données

`spec16`

Description

Enregistrement d'un comptage de particules sur fond d'émission constant.

Tâche

Ajuster un modèle approprié, déterminer ses paramètres, discuter la qualité de l'ajustement ainsi que les incertitudes.

Exercice 5

Données

`spec18`

Description

Enregistrement d'un spectre présentant deux raies en émission, l'une à profil gaussien, l'autre à profil lorentzien. L'instrument est caractérisé par un bruit gaussien.

Tâche

Ajuster un modèle approprié, déterminer ses paramètres, discuter la qualité de l'ajustement ainsi que les incertitudes.

MIDAS en quelques mots-clé

Quelques commandes courantes

`compute` *fichnom*␣=␣*opérations*

effectuer les *opérations* et écrire le résultat dans le fichier *fichnom*; attention aux blancs impératifs (␣) autour du signe '='

`compute` *opérations*

effectuer les *opérations* et afficher les résultats à l'écran

`create/graphics`

ouvrir une fenêtre graphique

`plot` *fichnom*

tracer le contenu du fichier *fichnom* à l'aide d'une courbe

`plot/pt` *fichnom*

tracer le contenu du fichier *fichnom* en n'affichant que les points

`plot/pl` *fichnom*

tracer le contenu du fichier *fichnom* en affichant les points reliés par des traits

`write` *fichdest* $abscissa,n$ $val_1,val_2,\ldots,val_n$

écrire n valeurs dans le fichier *fichdest* à partir de $x = abscissa$ (inclus); ne pas mettre de blanc(s) près des virgules de la liste

`overplot` *fichnom*

dessiner le contenu du fichier *fichnom* au-dessus du graphe existant

`read/descriptor` *fichnom*

afficher le descriptif du fichier *fichnom* à l'écran

`get/gcur` *fichnom*

basculer de la saisie clavier vers la saisie avec le curseur graphique pour permettre de pointer des valeurs dans le fenêtre graphique avec la souris (bouton de gauche); rebasculer vers la saisie clavier à l'aide d'un clic sur le bouton de droite ou du milieu.

`set/graphics color=`*idxcouleur*

définir la couleur d'indice *idxcouleur* pour l'opération graphique suivante (`plot`, `overplot`)

idxcouleur	couleur
0	blanc (invisible)
1	noir
2	rouge
3	vert
4	bleu
5	jaune
6	magenta (rouge-rose)
7	cyan (turquoise)
8	blanc (invisible)

`extract` *fichdest*␣=␣*fichnom*[x_1:x_2]

extraire la portion couvrant l'intervalle de x_1 à x_2 (inclus) du fichier *fichnom* et attribuer le résultat au fichier *fichdest*; attention aux blancs impératifs autour du signe '='

`stat` *fichnom*
`stat` *fichnom*[x_1:x_2]

afficher des informations statistiques (moyenne, déviation standard, moments, etc.) du fichier *fichnom*, soit en entier, ou, si spécifié, relatif à l'intervalle de x_1 à x_2 (inclus)

Quelques particularités

- Les **arguments des fonctions trigonométriques** sont à exprimer en degrés sous MIDAS; sous FORTRAN, comme en mathématiques en général, ils sont en radians. Le cas échéant, il faut donc veiller à interpréter correctement les valeurs de paramètres produits par des programmes FORTRAN, tels que `ajuste.f`, avant de les utiliser dans MIDAS.
- Le **logarithme Nepérien** se note `ln(`*arg*`)` sous MIDAS.

Quelques procédures supplémentaires

`tfd` *Real_f Imag_f Real_TF Imag_TF*

calcule la transformée de Fourier directe de la fonction f, de parties réelle *Real_f* et imaginaire *Imag_f*; la partie réelle de la transformée de f est retournée dans *Real_TF*, la partie imaginaire dans *Imag_TF*

`tfi` *Real_f Imag_f Real_IF Imag_IF*

calcule la transformée de Fourier inverse de la fonction f, de parties réelle *Real_f* et imaginaire *Imag_f*; la partie réelle de la transformée inverse de f est retournée dans *Real_IF*, la partie imaginaire dans *Imag_IF*

`expand` *fichnom fichdest n*

compléter le fichier *fichnom* de manière à ce qu'il contienne *n* points et sauver le résultat dans le fichier *fichdest*

`fortran source`

compiler le code FORTRAN `source.f` (p.ex.,`genfunc.f`, `ajuste.f`)

`genfunc` *fichdest*

générer une fonction (filtre, fenêtre) avec `genfunc.f` et sauver le résultat dans *fichdest*

`ajuste` *fichnom sigma fichmod* $a_1,a_2,\ldots$

ajuster le modèle défini dans ajuste.f aux données du fichier *fichnom*, en utilisant les barres d'erreur (bruit) *sigma* et les valeurs a_1, a_2, ... comme estimations des valeurs initiales des paramètres; attention: pas de blancs autour des virgules de la liste des paramètres.