

I. Des tableaux de données

Nous allons voir dans cette section comment saisir de petits tableaux de données de l'un des types suivants.

- Tableaux d'images de plusieurs fonctions à une seule variable.
- Tableaux d'images d'une seule fonction à deux variables.

Note. Tous les tableaux seront fabriqués via `\begin{funcstab}... \end{funcstab}` en utilisant un langage spécifique simplifiant la saisie des informations. Cet environnement est assez « malin » pour deviner le type de tableau souhaité en fonction des instructions fournies comme nous le constaterons dans cette documentation.

1. Tableaux d'images de fonctions à une seule variable

Les valeurs pivots, qui sont celles dont on veut indiquer les images, s'indiquent de deux façons.

1. `xvals = x_1 , x_2 , ... , x_n` indiquent les valeurs pour la variable nommée x par défaut.
2. `xvals = mavar : x_1 , x_2 , ... , x_n` permet de renommer la variable en *mavar*.

Une fois ceci fait, il faut renseigner les différentes images en donnant obligatoirement la formule de l'expression étudiée en passant via `imgs = monexpr : im_1 , im_2 , ... , im_n`.

En résumé, tout se passe via `xvals = ...` et `imgs = ...` pour définir les valeurs pivots et leurs images.

Exemple 1 (Une seule fonction avec la variable par défaut).

```
\begin{funcstab}
  xvals =      1      ,  2      ,  3      ;
  imgs  = f(x) : 1_{un} ,  2^2 ,  3^{3^3}
\end{funcstab}
```

—
CTXT: [xvals]

LABEL: []

VAL: (1, 2, 3)

—
CTXT: [imgs]

LABEL: [f(x)]

VAL: (1_{un}, 2², 3^{3³})

NEW DATA 1

TODO

Note. Retenir que tout se saisie en mode mathématique.

Danger. L'utilisation de `xvals` doit être faite une fois, et une seule, au tout début du contenu.

Exemple 2 (Deux fonctions pour une variable « maison »).

```
\begin{funcstab}
  xvals =      t : 10      ,  20      ,  30      ;
  imgs  = a(t) : a      ,  B \cdot B ,  \frac{c}{c} ;
          b(t) : x - y ,  \cos z      ,  t_0
\end{funcstab}
```

—
CTXT: [xvals]

LABEL: [t]

VAL: (10, 20, 30)

—
CTXT: [imgs]

```

LABEL: [a(t)]
VAL: (a, B · B,  $\frac{e}{c}$ )
—
CTXT: [imgs]
LABEL: [b(t)]
VAL: (x − y, cos z, t0)

```

NEW DATA 2

TODO

Exemple 3 (Commentaires à la sauce L^AT_EX).

```

% Commenter se fait avec parcimonie.

\begin{functable}
% Les valeurs de la variable x qui nous intéressent.
  xvals =      1 , 20 , 300 , 4000 ;
% Les images de x par la fonction f.
  imgs = f(x) : a , bb , ccc , dddd
\end{functable}

% Rien de plus à ajouter !

```

```

—
CTXT: [xvals]
LABEL: []
VAL: (1, 20, 300, 4000)

```

```

—
CTXT: [imgs]
LABEL: [f(x)]
VAL: (a, bb, ccc, dddd)

```

NEW DATA 3

TODO

Astuce (Nombres décimaux en version « locale » et « grandes » fractions). Via les macros `\dfrac` et `\num`¹ venant des excellents packages `amsmath` et `siunitx` respectivement, il est facile de rédiger des nombres décimaux, et d’obtenir de « grandes » fractions comme le montre l’exemple suivant.

```

% \usepackage{amsmath}
% \usepackage[locale=FR]{siunitx}

\begin{functable}
  xvals =      a , b      , c      ;
  imgs = f(x) : 1 , \num{2345678.90123} , \dfrac{45}{\num{67890}}
\end{functable}

```

```

—
CTXT: [xvals]
LABEL: []
VAL: (a, b, c)

```

```

—
CTXT: [imgs]
LABEL: [f(x)]
VAL: (1, 2 345 678,901 23,  $\frac{45}{67\,890}$ )

```

1. Cette macro ajoute de fins espaces mettant en valeur les groupes de chiffres, tout en gérant l’absence d’espaces autour du séparateur décimal.

NEW DATA 4

TODO