

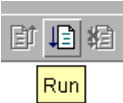
# Liste des commandes (non exhaustive) sous Matlab

## Cours 6 : Fichiers M sous Matlab – scripts et fonctions

### Création et édition de fichiers M

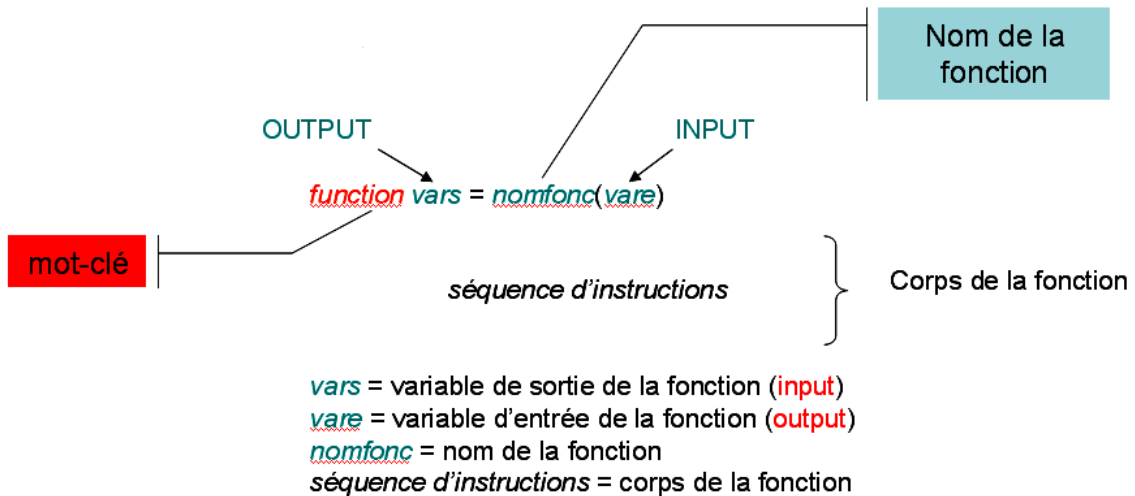
- **edit** : ouvre l'éditeur de texte
  - équivalent à **Menu File>New>M-File**
- **edit nomfich** : ouvre le fichier **nomfich.m** dans l'éditeur de texte
- **what** : affiche la liste des noms des fichiers du répertoire courant
- **type nomfich.m** : affiche dans la fenêtre de commandes le contenu du fichier **nomfich.m**

### Les scripts

- **Structure de base** :  
    %lignes d'aides  
    séquence d'instructions
- **Sauvegarder un script** : dans un fichier M
  - par exemple, **nomscript.m**
- **Exécution d'un script** :
  - dans l'éditeur de texte : **Menu Debug>Run** ou l'icône   
    Run
  - dans la fenêtre de commandes : taper le nom du fichier M  
    par exemple : **nomscript**

## Les fonctions

- **Définition d'une fonction :**



- **Structure de base :**

1. **Définition de la fonction** : une ligne qui définit le nom de la fonction, le nombre et l'ordre des inputs et des outputs.
2. **La ligne H1** : première ligne d'aide « help1 » recherchée quand on utilise la commande *look for*. Cette ligne commence par le caractère %.
3. **Le texte d'aide** : texte d'aide affiché à la suite de la ligne H1 quand on utilise la commande *help*. Chaque ligne du texte commence par le caractère %.
4. **Le corps de la fonction** : contient le code qui exécute les calculs et assigne des valeurs aux arguments outputs.
5. **Les commentaires** : une ligne de commentaires commence impérativement par le caractère % et peut apparaître à n'importe quel endroit dans le fichier.

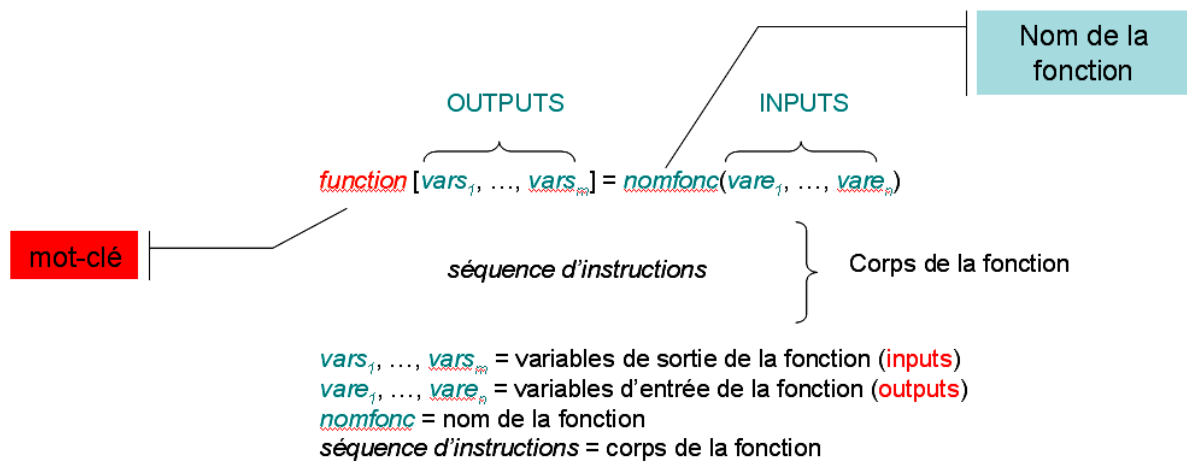
- **Sauvegarder une fonction :**

- enregistrer impérativement la fonction dans le fichier *nomfnc.m*
- le fichier M doit porter le nom de la fonction qu'il appelle

- **Exécuter une fonction** : appeler la fonction par son nom *nomfnc*

- affecter une valeur à la variable d'entrée *inp*
- *nomfnc(inp)* dans la fenêtre de commande
  - retourne le résultat dans la fenêtre de commande
- *out=nomfnc(inp)*
  - sauvegarde le résultat dans une variable *out*

- **Définition d'une fonction : multiples inputs/outputs**



- **Exécuter une fonction : multiples outputs**

- affecter une valeur aux variables inputs `inp1, ..., inpn`
- quand la fonction retourne plus d'un output :

`[out1, ..., outm] = nomfnc(inp1, ..., inpn)`

- retourne dans les variables `out1, ..., outm` les valeurs des outputs

`[out1, ..., outk] = nomfnc(inp1, ..., inpn)`

- retourne dans les variables `out1, ..., outk` les  $k$  premiers outputs

`nomfnc(inp1, ..., inpn)`

- retourne uniquement le premier output

- l'ordre d'appel des arguments input et output est important !
- ranger les variables de sortie (outputs) par ordre d'importance

- **Définition d'une fonction : pas d'outputs**

`function nomfnc(vare1, ..., varen)`  
`function [] = nomfnc(vare1, ..., varen)`

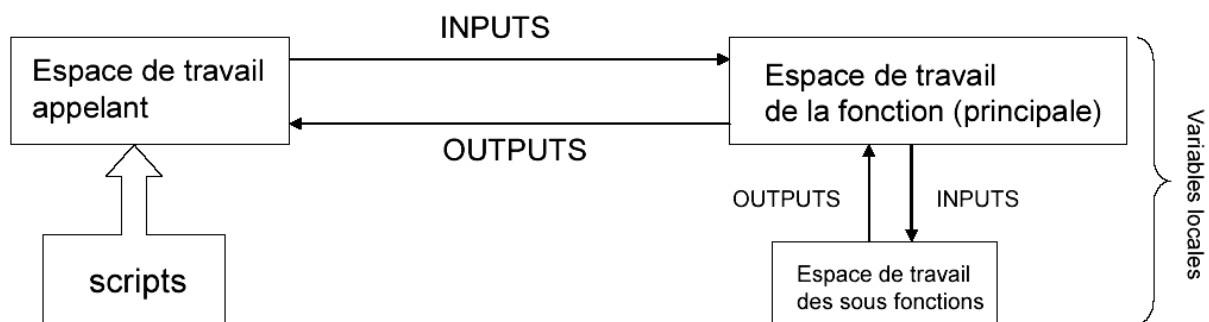
## Les sous fonctions

- **Définition :**
  - fonctions contenues dans le code d'une fonction principale
  - ordre d'apparition indifférent des sous fonctions
- **Identification de la nature d'une fonction (par ordre de priorité) :**
  1. une sous fonction
  2. une fonction privée
  3. une fonction standard du « search path »

## Les fonctions privées

- **Définition :**
  - fonctions se trouvant dans un sous répertoire appelé « *private* »
  - uniquement accessibles aux fonctions du répertoire parent contenant le sous répertoire « *private* »

## Espace de travail des fonctions



## Variables locales

- **Variables locales dans une fonction :**
  - Variables définies dans le corps de la fonction
  - Variables propres à chaque fonction qui les utilisent
    - ne restent pas en mémoire après l'exécution de la fonction
- **Variables locales dans un script :**
  - Variables définies dans le corps d'un script
  - Variables stockées dans l'espace de travail partagé avec le programme appelant

## Variables globales

- **Définition** :
  - variable dont la valeur peut être partagée par plusieurs fonctions
- **Déclaration des variables globales** :  
  
**function** [*vars*<sub>1</sub>, ..., *vars*<sub>m</sub>] = *nomfnc*(*vare*<sub>1</sub>, ..., *vare*<sub>n</sub>)  
*global* *VAR*<sub>1</sub> *VAR*<sub>2</sub> ... *VAR*<sub>N</sub>
  - *VAR*<sub>1</sub> *VAR*<sub>2</sub> ... *VAR*<sub>N</sub> = variables globales
  - par convention, déclarées en début de fichier après la ligne de définition et en majuscules
- **who global** : affiche le nom des variables globales de l'espace de travail
- **whos global** : affiche le nom et les informations détaillées des variables globales
- **clear global** : supprime toutes les variables globales
- **clear global** *VAR*<sub>1</sub>, *VAR*<sub>2</sub>, ..., *VAR*<sub>N</sub> : supprime les variables globales *VAR*<sub>1</sub>, *VAR*<sub>2</sub>, ..., *VAR*<sub>N</sub>

## Variables persistantes

- **Définition** :
  - variable dont la valeur est seulement accessible à la fonction dans laquelle elle est déclarée
  - variable locale à la fonction mais dont la valeur reste stockée en mémoire
- **Déclaration des variables persistantes** :  
  
**function** [*vars*<sub>1</sub>, ..., *vars*<sub>m</sub>] = *nomfnc*(*vare*<sub>1</sub>, ..., *vare*<sub>n</sub>)  
*persistent* *VAR*<sub>1</sub> *VAR*<sub>2</sub> ... *VAR*<sub>N</sub>
  - *VAR*<sub>1</sub> *VAR*<sub>2</sub> ... *VAR*<sub>N</sub> = variables persistantes
  - par convention, déclarées en début de fichier après la ligne de définition et en majuscules