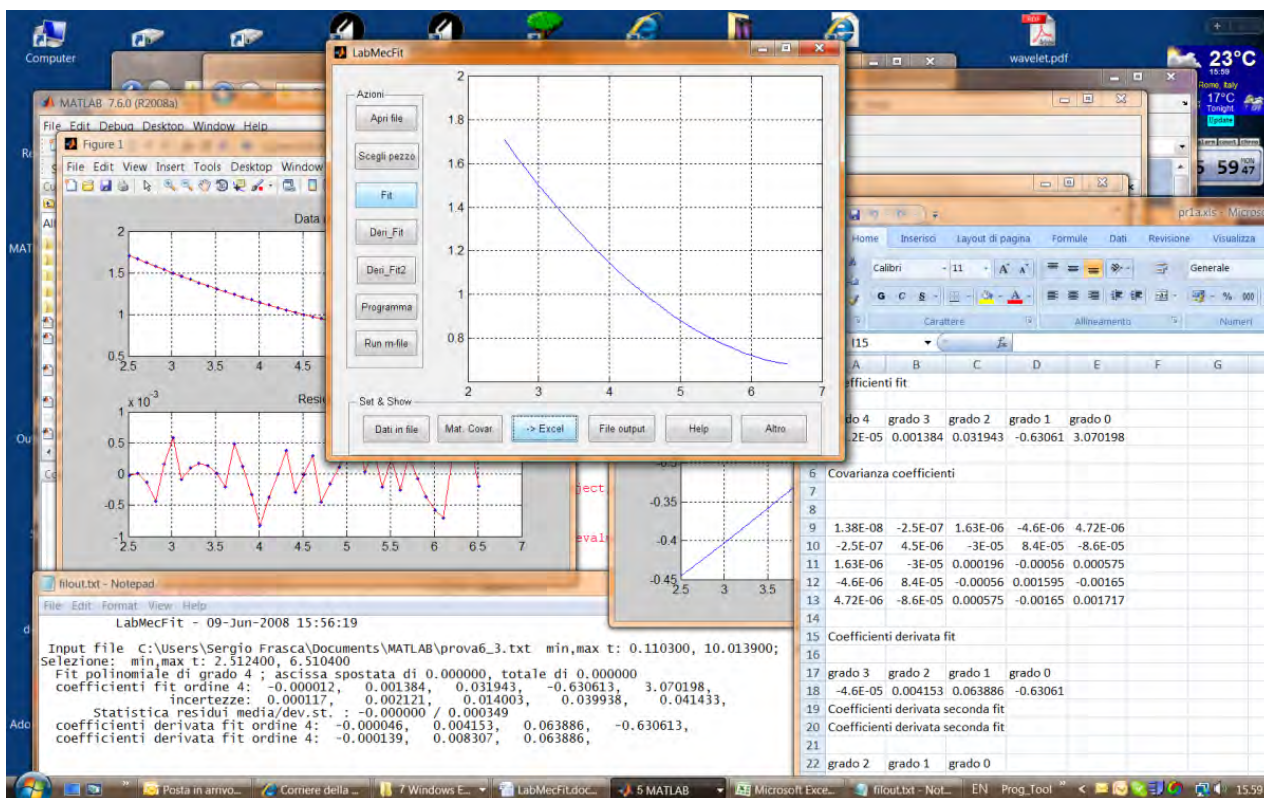


# LabMecFit

versione beta

by S.Frasca – Dipartimento di Fisica Università Sapienza – Roma

LabMecFit è un programma che permette di elaborare i dati prodotti da DataStudio. I dati devono essere salvati da DataStudio nel formato testo esportabile.



## Sommario

Installazione.....	2
Esecuzione.....	3
Approfondimenti.....	5

## Installazione

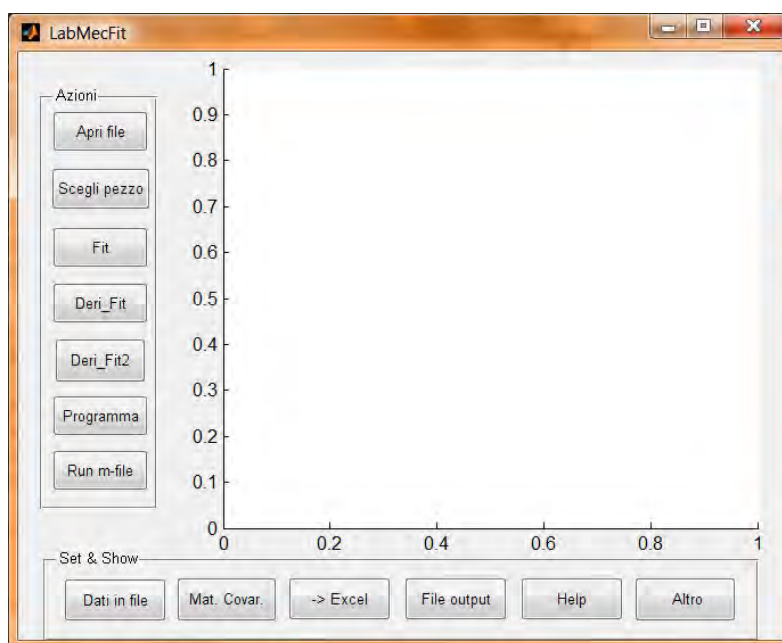
LabMecFit è un programma Matlab che può essere eseguito all'interno del sistema Matlab o come applicazione standalone.

Per essere usato all'interno del sistema Matlab occorre installare Snag (vedi <http://grwavsf.roma1.infn.it/snag/>), o almeno le m-funzioni richieste.

Per essere usato standalone, occorre installare il runtime di Matlab, della stessa versione usata per realizzare l'eseguibile. In appendice ci sono le indicazioni per i vari sistemi operativi. Notare in particolare che c'è differenza tra Windows 32 bit e Windows 64 bit.

## Esecuzione

Al lancio di LabMecFit appare una finestra di testo e la seguente finestra:



Le operazioni base sono le seguenti:

- **Apri file** , in cui si sceglie il file di dati (tipicamente .txt) che contiene i dati che vogliamo analizzare
- **Scegli pezzo** , in cui si sceglie il segmento di dati da analizzare; questa operazione può farsi in più passi, ogni volta restringendo il segmento
- **Fit** , calcola il fit polinomiale dei dati, i residui e la derivata prima e seconda dei dati; viene richiesto il grado del polinomio ed eventualmente il valore di cui si vuole ridurre l'ascissa (vedi approfondimenti). Viene aperta una finestra con due grafici: i dati selezionati (punti) col fit (in rosso) e i residui del fit. Viene anche calcolata la matrice di covarianza dei coefficienti del fit (e quindi le incertezze su di essi).

La derivata prima e seconda dei dati possono essere graficate spingendo i due bottoni relativi.

Al lancio del programma si crea un file (filout.txt) in cui va traccia di tutte le operazioni eseguite. Il nome del file può essere cambiato (nel nuovo file andranno tutte e sole le operazioni successive).

Ecco un esempio:

LabMecFit - 09-Jun-2008 15:56:19

Input file C:\Users\Sergio Frasca\Documents\MATLAB\prova6\_3.txt min,max t: 0.110300, 10.013900; senserror = 0.001000

Selezione: min,max t: 2.512400, 6.510400

Fit polinomiale di grado 4 ; ascissa spostata di 0.000000, totale di 0.000000

coefficienti fit ordine 4: -0.000012, 0.001384, 0.031943, -0.630613, 3.070198,

incertezze: 0.000117, 0.002121, 0.014003, 0.039938, 0.041433,

Statistica residui media/dev.st. : -0.000000 / 0.000349

coefficienti derivata fit ordine 4: -0.000046, 0.004153, 0.063886, -0.630613,

coefficienti derivata seconda fit ordine 4: -0.000139, 0.008307, 0.063886,

In questo file si possono mettere, spingendo i relativi bottoni della finestra principale, anche i dati selezionati e quelli da questi calcolati col fit e la matrice di covarianza.

Tutti i dati calcolati possono andare, a richiesta, in un file Excel (ovviamente se Excel è presente sul calcolatore ospite).

C'è anche la possibilità di dare dei comandi Matlab da far eseguire. Ciò può farsi spingendo il pulsante "Programma" e dando i comandi nella finestra che appare. Le variabili che contengono i dati sono:

- t il tempo relativo alle misure selezionate
- s le misure selezionate
- fit il fit polinomiale fatto
- res i residui
- d1 la derivata prima del fit
- d2 la derivata seconda del fit
- p i coefficienti del polinomio di fit
- p1 i coefficienti del polinomio della derivata del fit
- p2 i coefficienti del polinomio della derivata seconda di fit

Per esempio il seguente programmino grafica il quadrato di d1 in funzione di s:

```
figure
plot(s,d1.^2)
grid on
```

## Approfondimenti

## Appendice

### MATLAB Deployment Checklist

=====

Use this checklist to successfully deploy your application, component, or library.

You can distribute a MATLAB Compiler-generated standalone application, component, or library to any target machine with the same operating system as the machine on which the application was compiled (the "source" development machine).

For more information, see the documentation for the MATLAB Compiler at: <http://www.mathworks.com/access/helpdesk/help/toolbox/compiler/>

NOTE: Replace the items enclosed in angle brackets ( < > ) with your own values.

1. Ensure that the MATLAB Compiler Runtime (MCR) is installed on target machines, and ensure you have installed the correct version. To verify this, do the following:

A. Attempt to verify the MCR exists on your system.

The MCR usually resides in these locations on these platforms:

Operating System	Path to MCR
=====	=====
Windows	C:\Program Files\MATLAB\MATLAB Compiler Runtime\v78
Mac	Applications/MATLAB/MATLAB_Compiler_Runtime/v78
Linux/Solaris	*** No predetermined installation location *** Contact your system administrator.

NOTE: For information about the installation process and the MCR, type "help MCR" or "help mcrinstaller" at the MATLAB command prompt.

B. MATLAB 2008a uses MCR version 7.8. To verify the version number of the installed MCR, type the following command:

```
[mcrmajor,mcrminor]=mcrversion
```

at the MATLAB command prompt.

2. Add the MCR directory to the path specified by the target system's environment variable.

A. Locate the name of the environment variable to set, using the table below:

Operating System	Environment Variable
=====	=====
Windows	PATH
Linux	LD_LIBRARY_PATH
Solaris	LD_LIBRARY_PATH
Mac	DYLD_LIBRARY_PATH

B. Set the path by doing one of the following:

On Windows systems:

\* Add the MCR directory to the environment variable by opening a command prompt and issuing the DOS command, specifying either win32 or win64:

```
set PATH=C:\Program Files\MATLAB\MATLAB Compiler Runtime\v78\runtime\{win32|win64};%PATH%
```

Alternately, for Windows, add the following pathname:

C:\Program Files\MATLAB\MATLAB Compiler Runtime\v78\runtime\{win32|win64}  
to the PATH environment variable, specifying either win32 or win64,  
by doing the following:

1. Select the My Computer icon on your desktop.
2. Right-click the icon and select Properties from the menu.
3. Select the Advanced tab.
4. Click Environment Variables.

On UNIX systems:

\* Add the MCR directory to the environment variable by issuing the following commands:

Linux

```
setenv LD_LIBRARY_PATH  
<mcr_root>/v78/runtime/glnx86:  
<mcr_root>/v78/sys/os/glnx86:  
<mcr_root>/v78/sys/java/jre/glnx86/jre1.6.0/lib/i386/native_threads:  
<mcr_root>/v78/sys/java/jre/glnx86/jre1.6.0/lib/i386/server:  
<mcr_root>/v78/sys/java/jre/glnx86/jre1.6.0/lib/i386  
setenv XAPPLRESDIR <mcr_root>/v78/X11/app-defaults
```

Solaris64

```
setenv LD_LIBRARY_PATH /usr/lib/lwp:  
<mcr_root>/v78/runtime/sol64:  
<mcr_root>/v78/sys/os/sol64:  
<mcr_root>/v78/sys/java/jre/sol64/jre1.6.0/lib/sparcv9/native_threads:  
<mcr_root>/v78/sys/java/jre/sol64/jre1.6.0/lib/sparcv9/server:  
<mcr_root>/v78/sys/java/jre/sol64/jre1.6.0/lib/sparcv9  
setenv XAPPLRESDIR <mcr_root>/v78/X11/app-defaults
```

Linux x86-64

```
setenv LD_LIBRARY_PATH  
<mcr_root>/v78/runtime/glnxa64:  
<mcr_root>/v78/sys/os/glnxa64:  
<mcr_root>/v78/sys/java/jre/glnxa64/jre1.6.0/lib/amd64/native_threads:  
<mcr_root>/v78/sys/java/jre/glnxa64/jre1.6.0/lib/amd64/server:  
<mcr_root>/v78/sys/java/jre/glnxa64/jre1.6.0/lib/amd64  
setenv XAPPLRESDIR <mcr_root>/v78/X11/app-defaults
```

Mac

```
setenv DYLD_LIBRARY_PATH  
<mcr_root>/version/runtime/maci:  
<mcr_root>/version/sys/os/maci:  
<mcr_root>/version/bin/maci:  
/System/Library/Frameworks/JavaVM.framework/JavaVM:  
/System/Library/Frameworks/JavaVM.framework/Libraries  
setenv XAPPLRESDIR <mcr_root>/version/X11/app-defaults
```

NOTE: To make these changes persistent after logout on UNIX or

Mac machines, modify the .cshrc file to include this setenv command.

NOTE: On Windows, the Environment Variable syntax utilizes backslashes (\), delimited by semi-colons (;). On UNIX, the EV syntax utilizes forward slashes (/), delimited by colons (:).

NOTE: When deploying C and C++ standalone applications, it is possible to run the shell script file run\_lab\_plot.sh on UNIX and Mac instead of setting environment variables. See "Requirements for Standalone Applications," below.

3. Collect the following files, based on what you are deploying:

Requirements for Standalone Applications:

=====

All Platforms: