Installation notes

In recent years MATLAB undertook several important changes, which made difficult for us to keep FSDA aligned to the MATLAB system. For example, from R2012b a Toolstrip replaced menus and toolbars in the MATLAB Desktop, a gallery of apps was introduced in the desktop and the documentation system was redesigned, for the first time after years of stability; with R2014a third party software documentation was moved to a separate section without filtering and searching possibilities; with R2014b there was a major update of the MATLAB graphics system that forced us to revisit most of the FSDA plotting functions; with R2015a the third party documentation was changed again, and was partly reintegrated in the MATLAB documentation system. This is why you may find these installation notes rather intricate: we apologise if this is going to happen and we invite you to signal problems or bugs that you think are in contradiction with, or should be reported in, these notes. With your help, it will be easier for us to make these installation notes … superfluous.

1. FSDA works from the release R2009b of MATLAB and uses the Statistics toolbox.
2. FSDA can be installed:
    a. Automatically with our setup program for Windows platforms. The automatic installation updates your MATLAB search path and installs the FSDA documentation pages in:

       `(docroot folder)\FSDA`, If your release of MATLAB is <=2012a

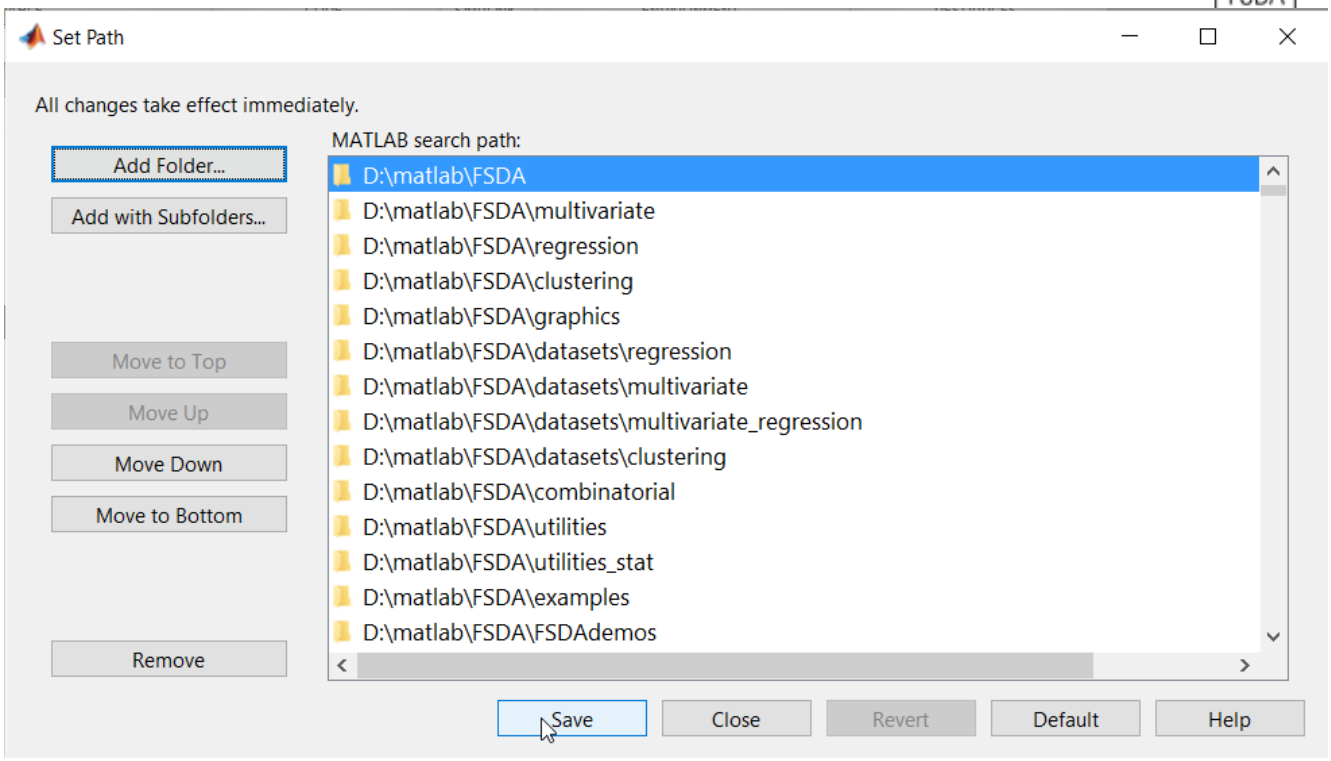
       `(FSDA root folder)\helpfiles\FSDA`, if your release of MATLAB is >2012a.

       To find out where your `docroot` folder is located it is enough o type `docroot` in the command prompt.

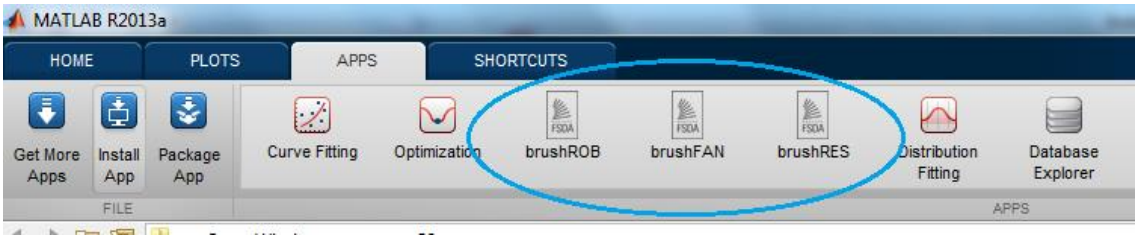
    b. Manually by unpacking the compressed tar file `FSDA.tar.gz` under a folder of your choice (say `programs`). The search path update can be done by running the MATLAB scripts `addFSDA2path.m` that is located in the `FSDA root folder`. If your release is >2012a the files present inside `(FSDA root folder)\helpfiles\FSDA` must be moved to `(docroot folder)\FSDA`.
3. If there are multiple releases of MATLAB installed in your computer, our setup program will ask you to **choose to which release the FSDA Toolbox has to be associated and where** (under which folder) it has to be installed. The search path update and documentation setup are modified accordingly. However, if other MATLAB releases are present and the user intends to run FSDA also on them, the two steps should be completed manually by using the already mentioned `addFSDA2path.m` (see 2b) as follows (now assuming a MS Windows platform installation under `D:\programs\FSDA`):

```
>> addFSDA2path('D:\programs\FSDA')
```

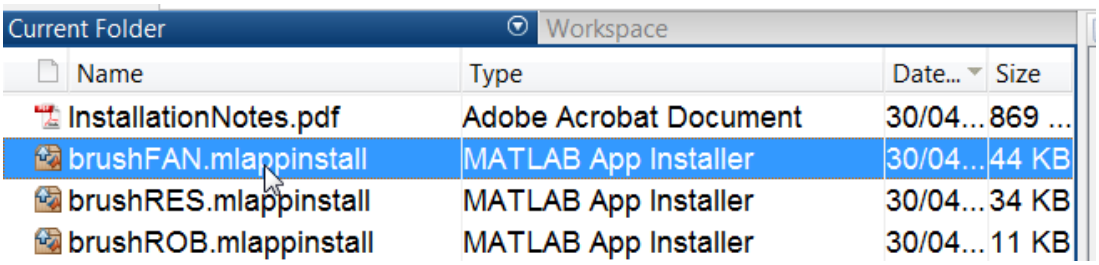
4. If FSDA has been installed properly (in what follows without loss of generality we assume, for example, that FSDA has been installed in folder `D:\matlab\FSDA`), after the installation the **"Set Path" window of MATLAB should include the following FSDA search paths** (the last three being introduced with FSDA V3.0 (i.e. with MATLAB R2015a).



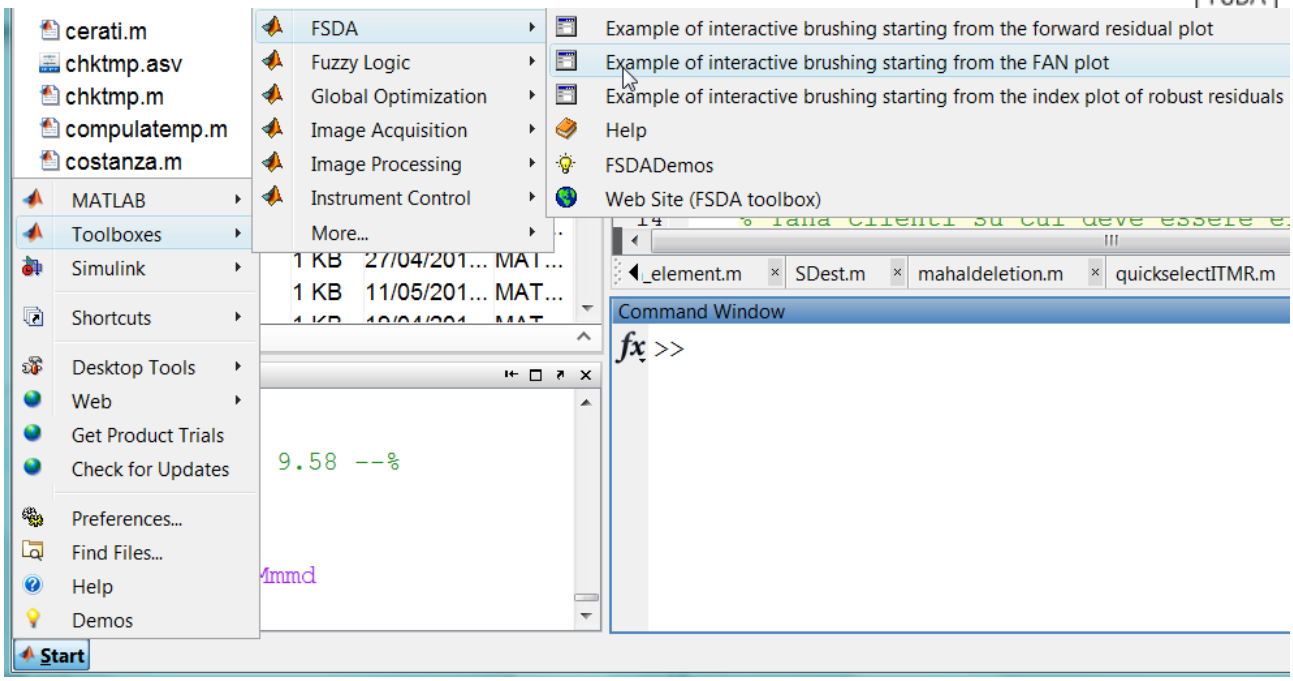
5. If FSDA is installed in MATLAB R2012b or subsequent releases, three APPS (`brushRES`, `brushFAN` and `brushROB`) are automatically installed:



Remark: if the three APPS have not been automatically installed, you can easily install them manually by double clicking on the files `brushFAN.mlappinstall`, `brushRES.mlappinstall` and `brushROB.mlappinstall` contained in the root folder of FSDA.

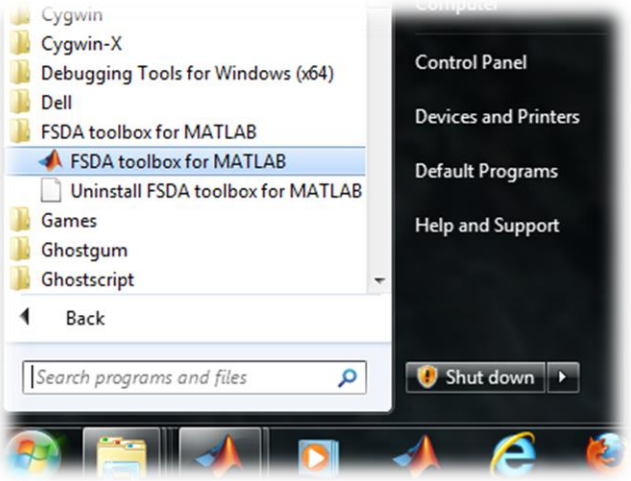


If FSDA is installed in MATLAB 2012a or earlier the three APPS appear inside `MATLAB Start button|Toolboxes|FSDA`.



These APPS are graphical user interfaces conceived to demonstrate some functionalities of FSDA.

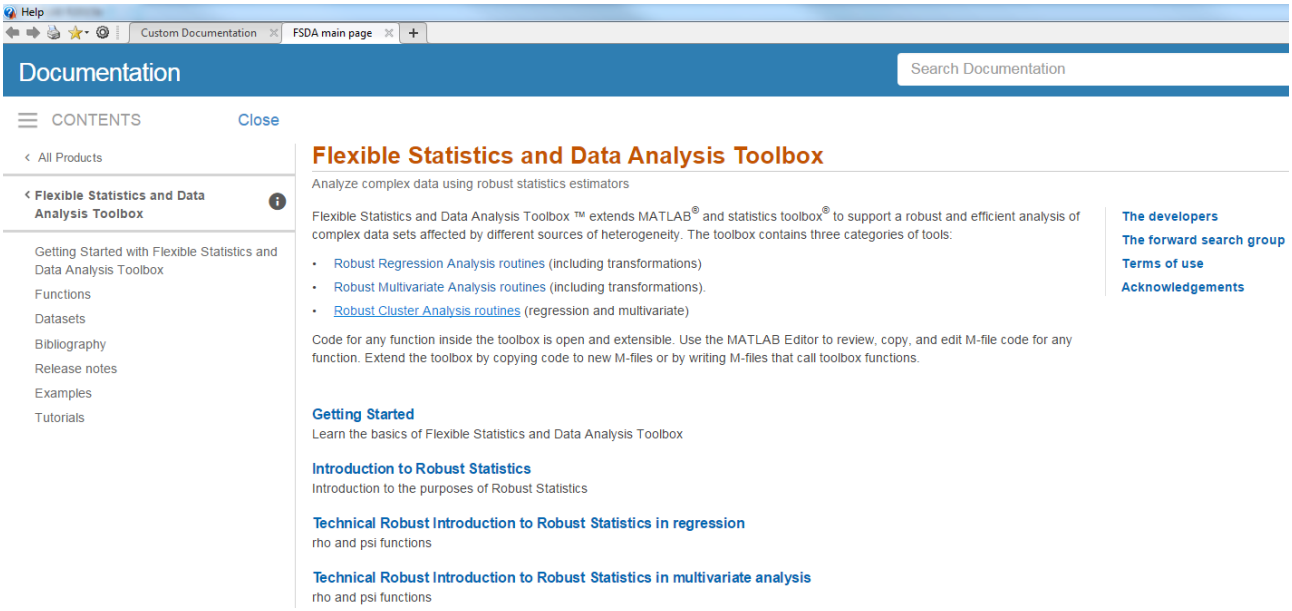
6. Our setup program, if successfully executed, adds to the "Program Files" Windows Menu the entry "FSDA toolbox for MATLAB", including a **FSDA uninstall program** that should be used by the user to remove an obsolete FSDA release, before an update:



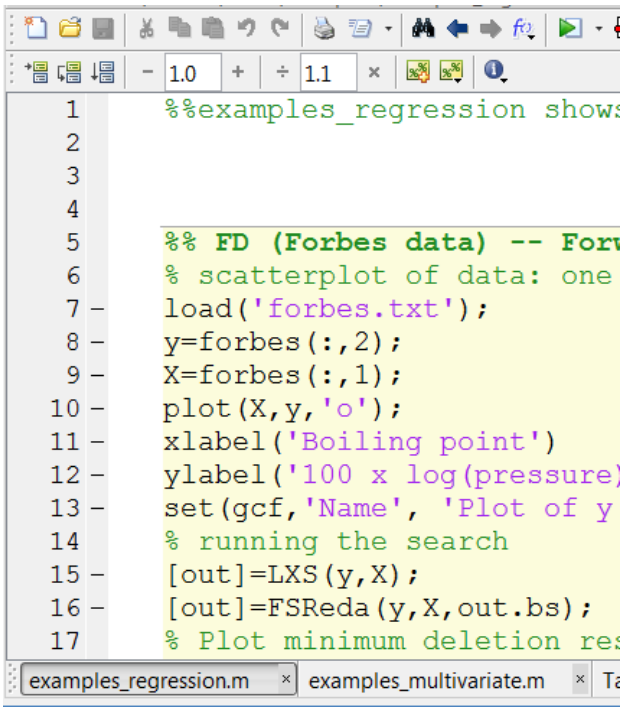
7. Nonetheless, to avoid problems that may occur if FSDA is installed with our setup program more than once, the setup program tries to locate and remove (with the agreement of the user) previous FSDA installations.

8. If everything went well with an automatic or manual installation, when you open MATLAB, the MATLAB "Help" pages should include FSDA with all its submenus.

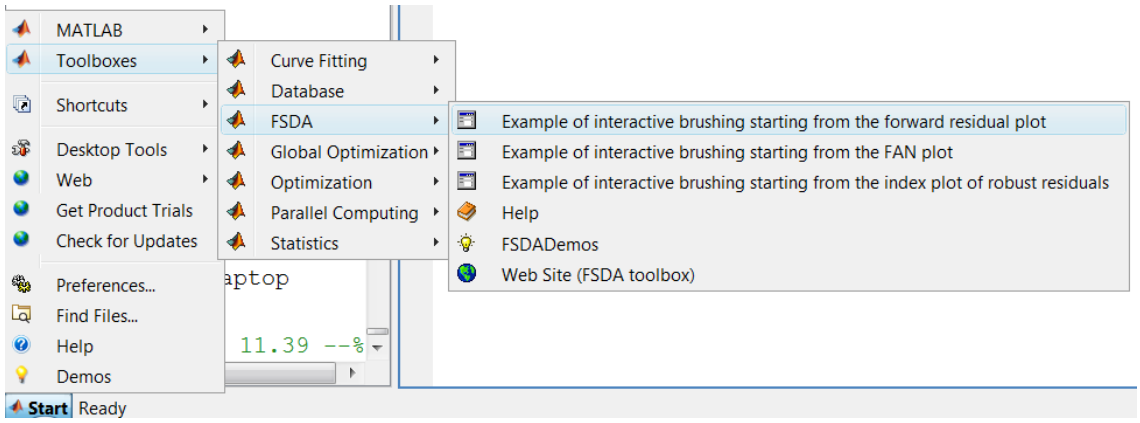
a. **The MATLAB "Help" pages should include FSDA**, as shown below:



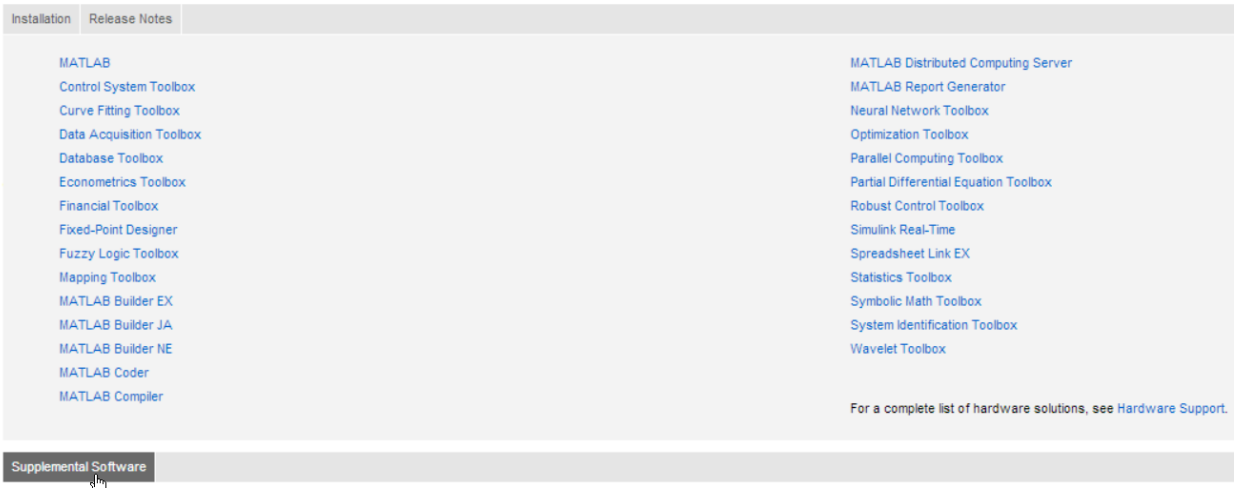
b. **With the setup installer, two example files named "examples_regression.m" and "examples_multivariate.m" should be opened automatically.** These files contain a series analysis of several well-known datasets in the literature of robust statistics and have the purpose to let the user familiarize with the toolbox (these two files are contained in (`main folder of FSDA)\examples`).



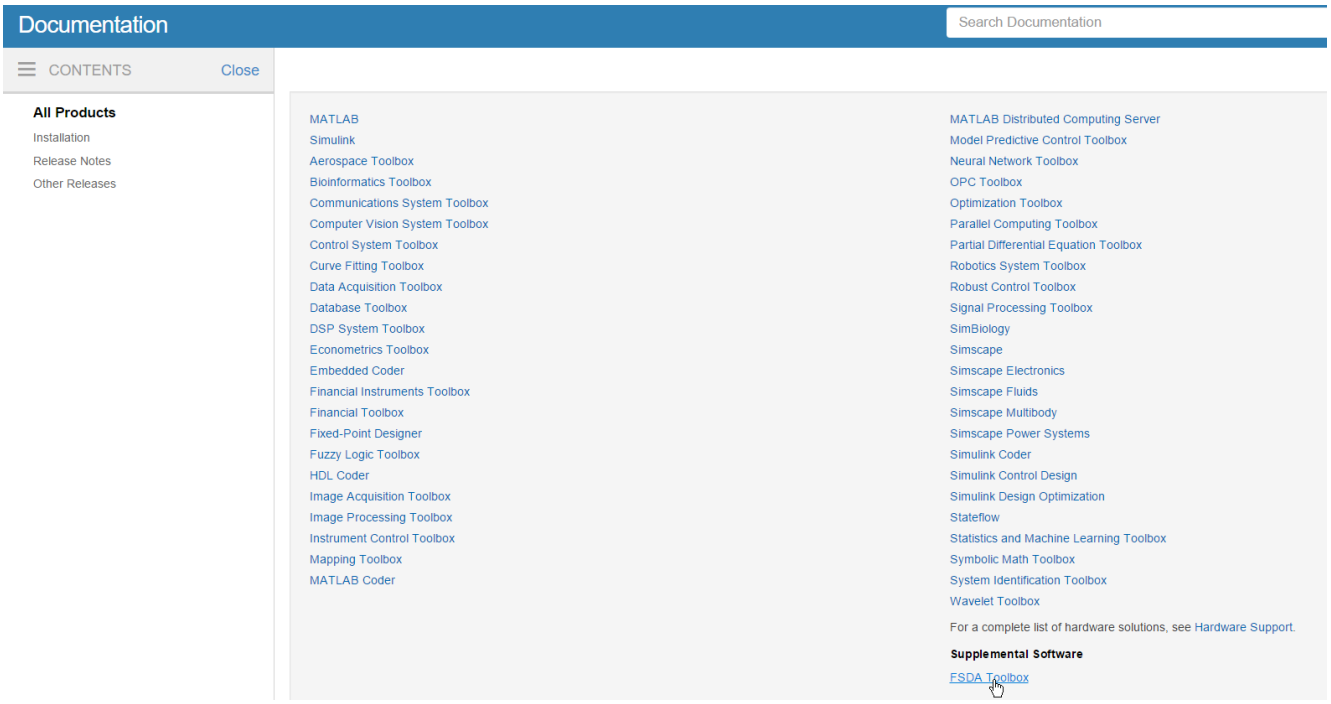
c. FSDA should appear among the installed "Toolboxes" in the MATLAB "Start Menu" (only for MATLAB releases before R2012b)



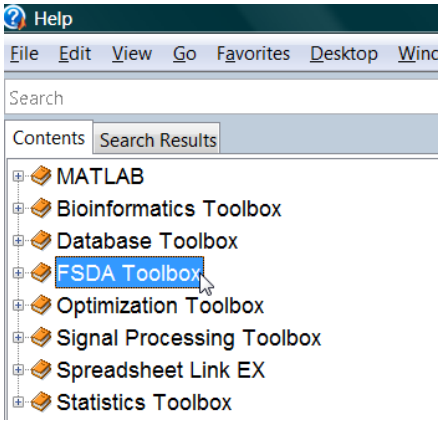
d. For MATLAB R2012b to R2014b installations, the html help files can be found in the Supplemental Software tab which appears at the bottom of the Doc Center home page (see screenshot below)**.**



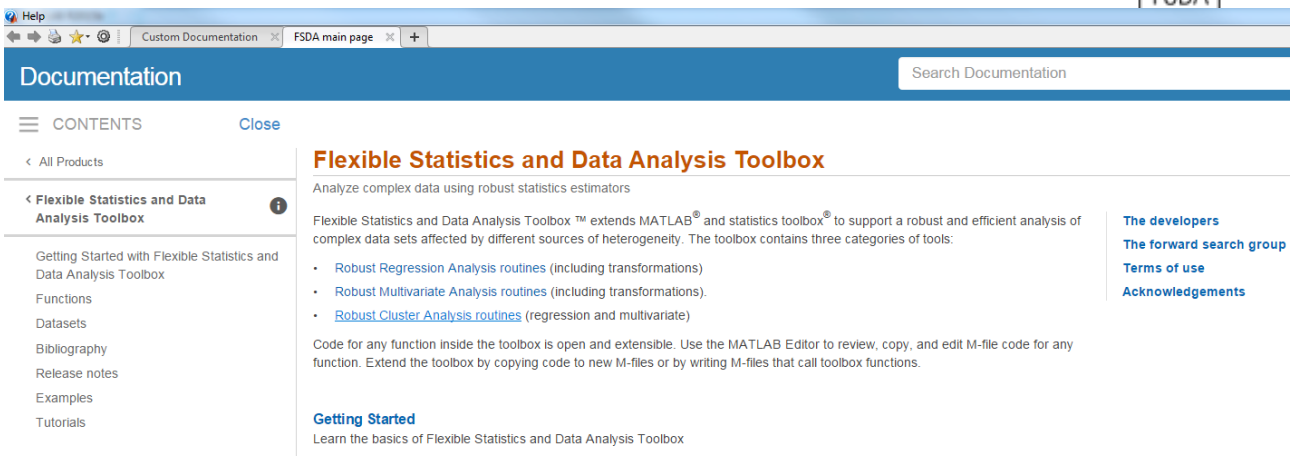
For MATLAB 2015a-2016b installations, the html help files can be found in the Supplemental Software link, (see screenshot below)



e. For MATLAB installations earlier than 2012b, the documentation is located in the same place as all the other official Mathworks toolboxes (see bottom panel of screenshot below):



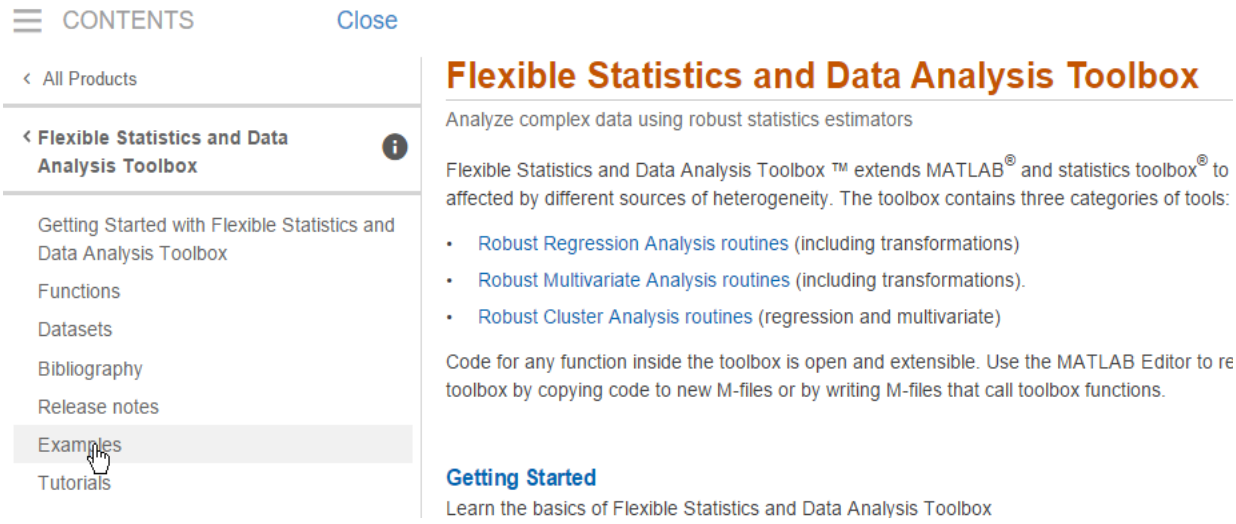
Independently from MATLAB version you use, once you click on the link FSDA Toolbox you should reach our main documentation page (see screenshot below)



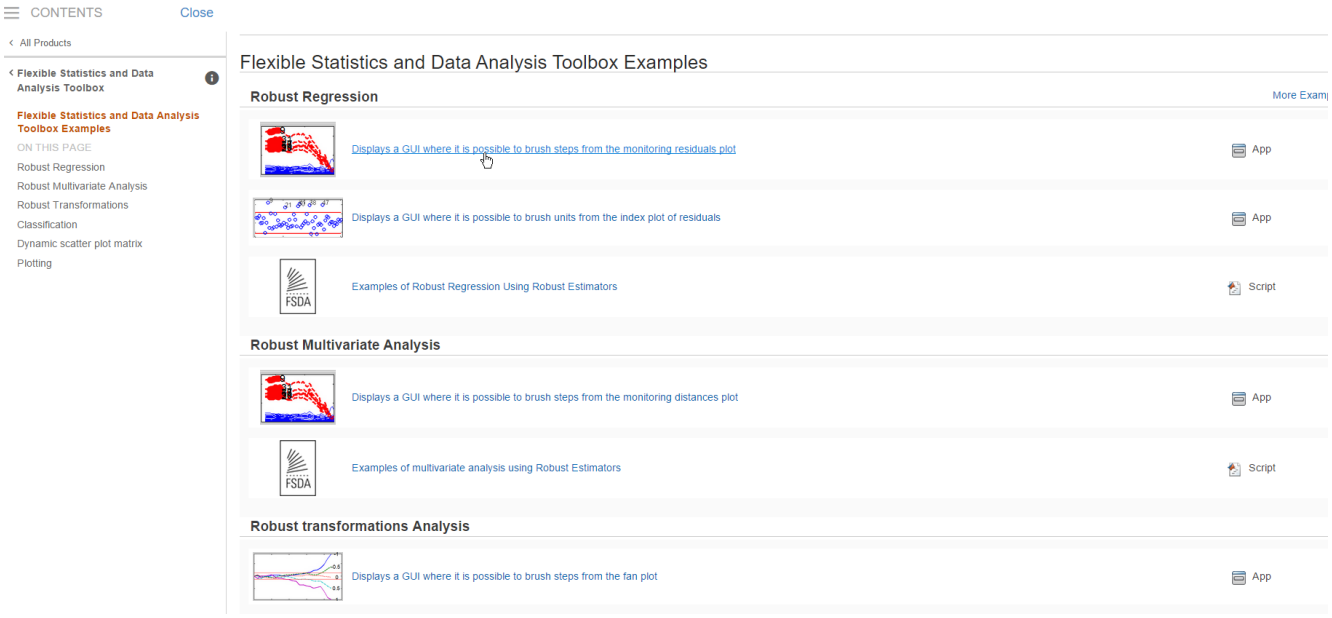
Remark: you can reach our main documentation page also simply typing `docsearchFS` in the command prompt

```
>> docsearchFS
```

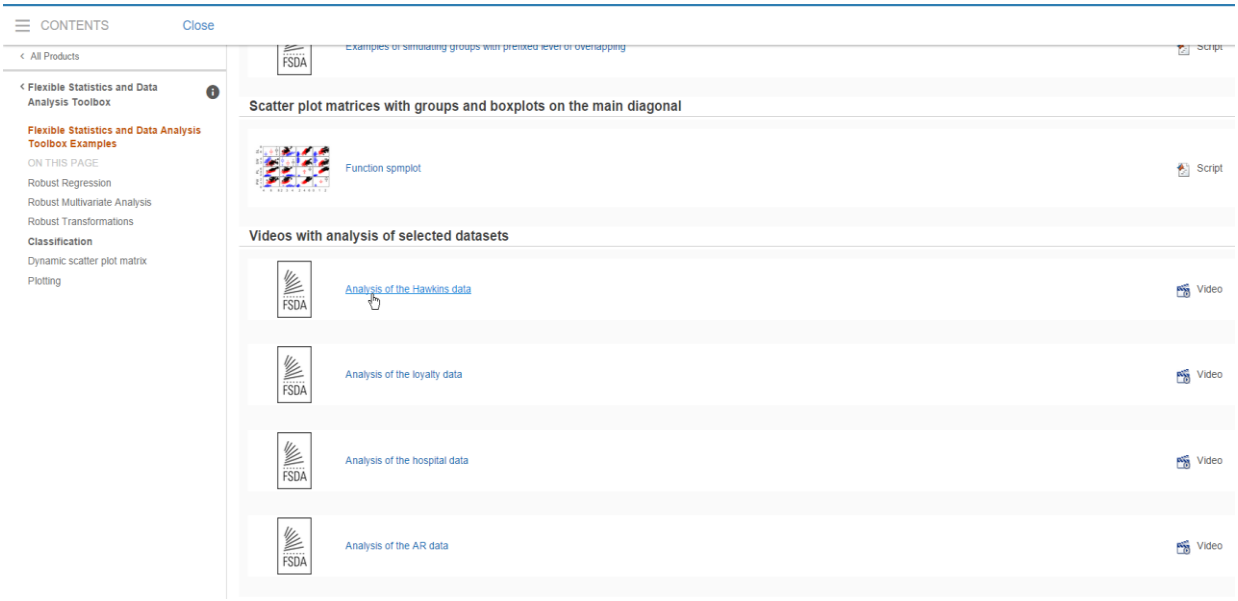
From our main documentation page you can go to the Examples page (see screenshot below),



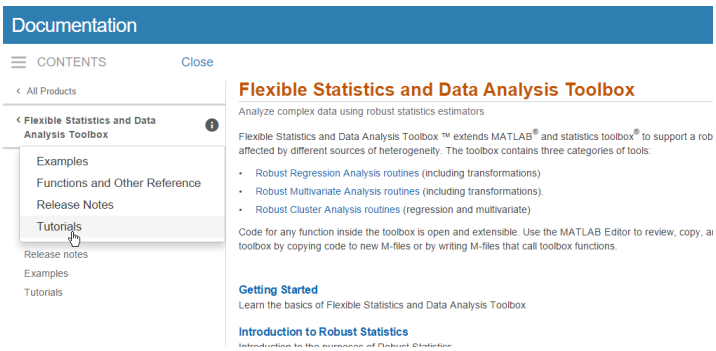
where you can find GUIs, example codes (see screenshot below).



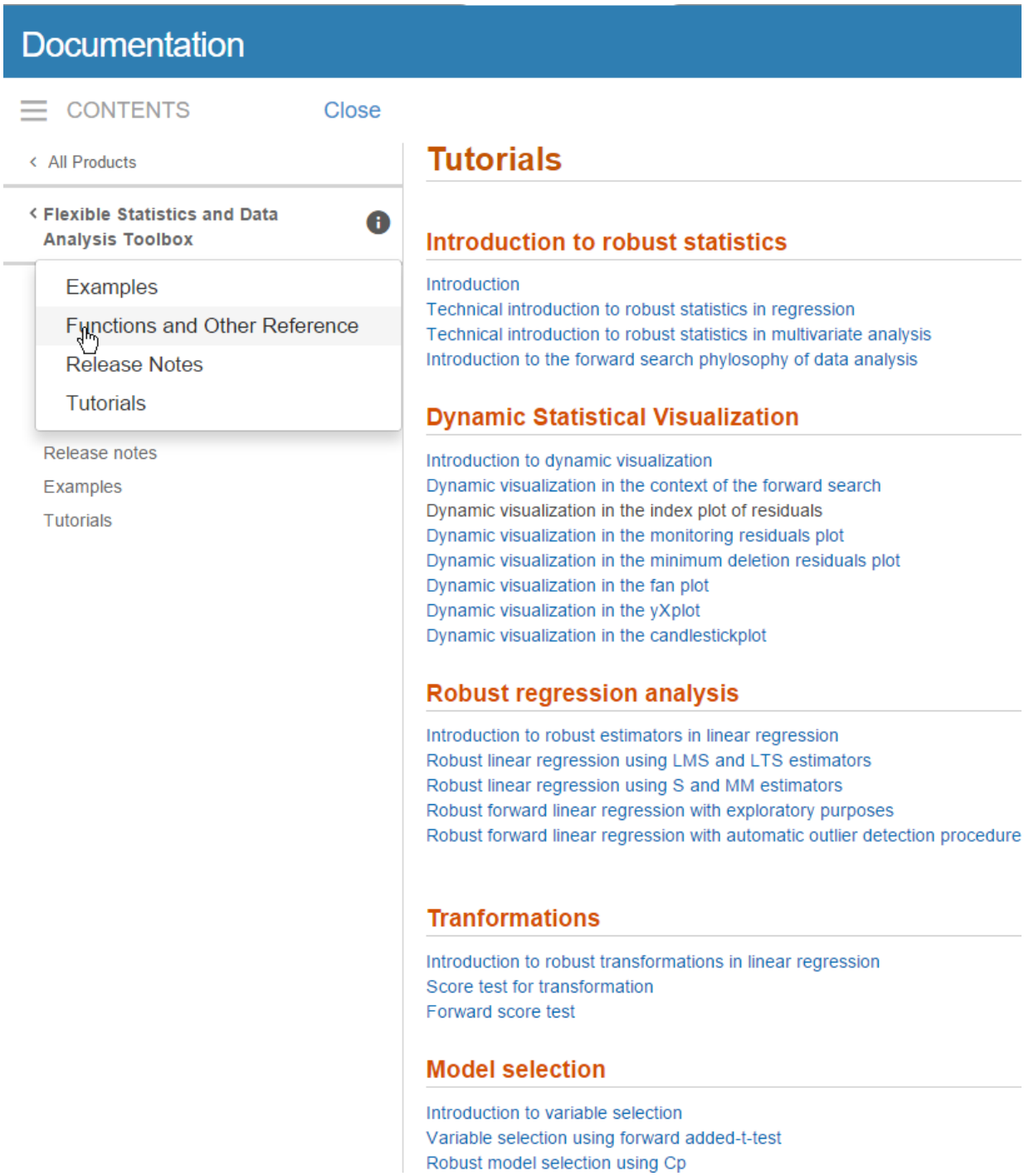
and links to videos containing the analysis of selected examples (see screenshot below)



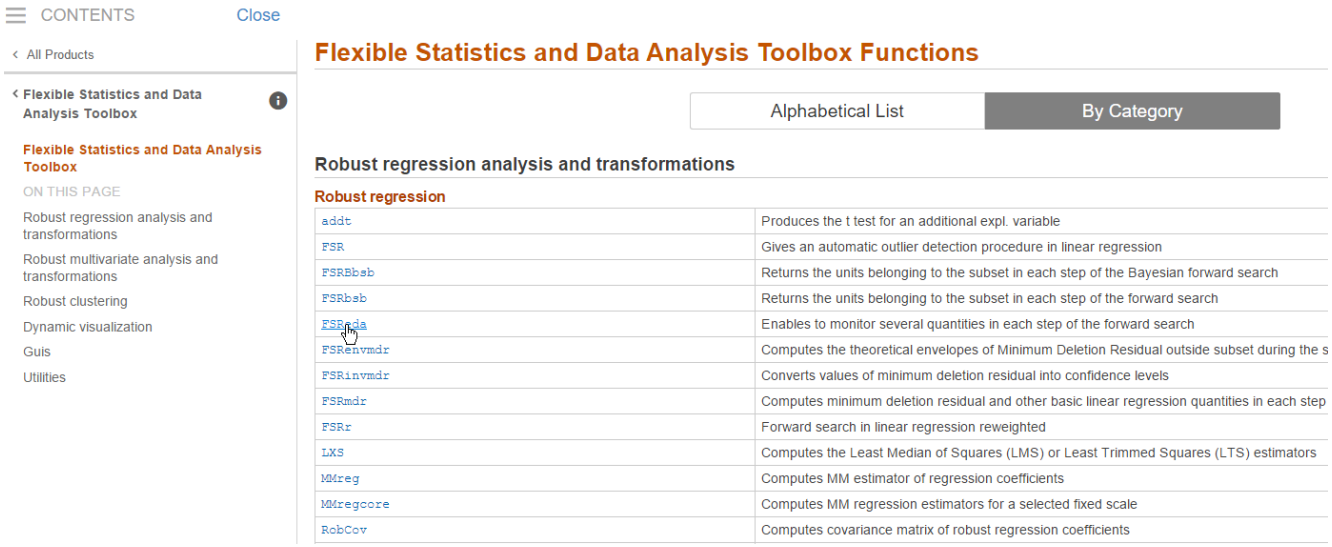
From any point of our documentation system you can go to the “Tutorials” page (see screenshot below)

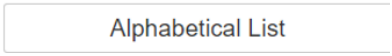


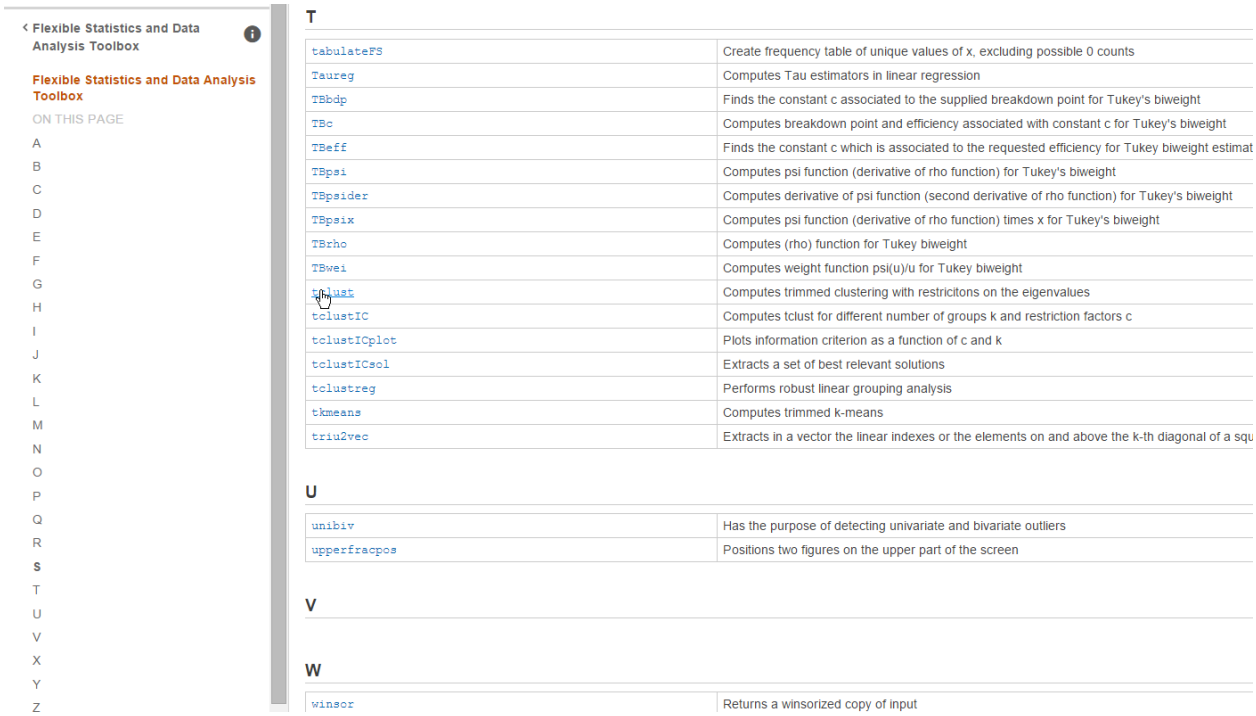
where you can find several tutorials about robust statistics and dynamic statistical visualization, transformations…. (see screenshot below).



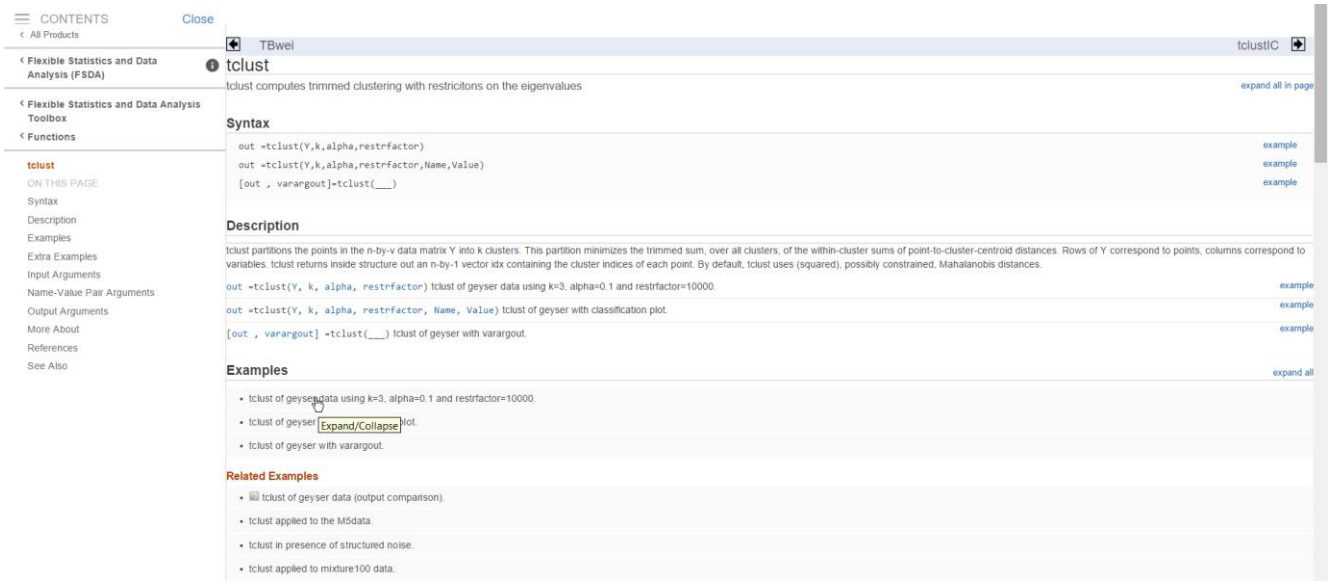
On the other hand, if from the left menu one clicks on “Functions and Other References” (see screenshot above), it is possible to go the categorical list of functions present in the toolbox (see screenshot below)



Of course clicking on the button  it is possible to browse in alphabetical order the documentation of the 165 functions present inside the FSDA toolbox (see screenshot below).

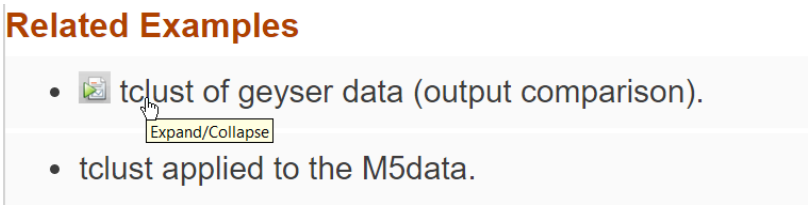


By clicking on one of these links (for example on `tclust`, see screenshot above) it is possible to reach the HTML documentation of the function in a perfect new MATLAB documentation style (see screenshot below)



These HTML documentation pages have been created automatically by our routines publishFS. Every HTML documentation contains a series of **Examples** and Related Examples.

The icon  at the beginning of the line, indicates that the associated example has been executed and its output has been captured inside the HTML file. For example, if you click on the first of the Related Examples (see screenshot below)



It is possible to see both the code (note that the code is displayed inside HTML using typical Matlab colouring) and the output which was generated (see the two screenshots below)

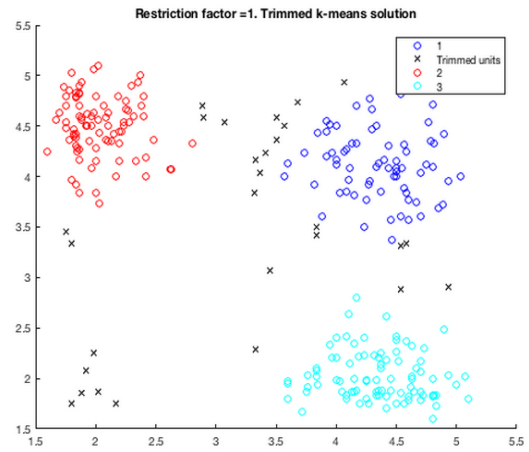
- tclust of geyser data (output comparison).

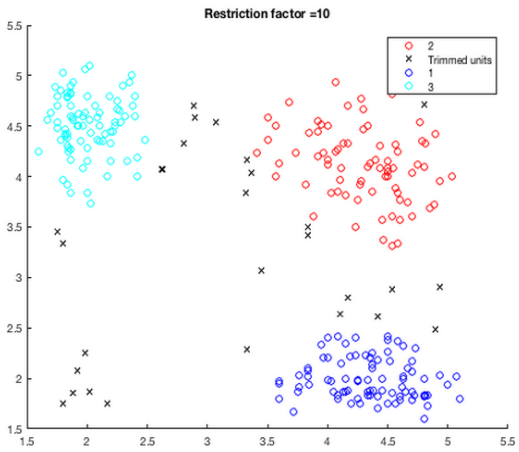
Expand/Collapse

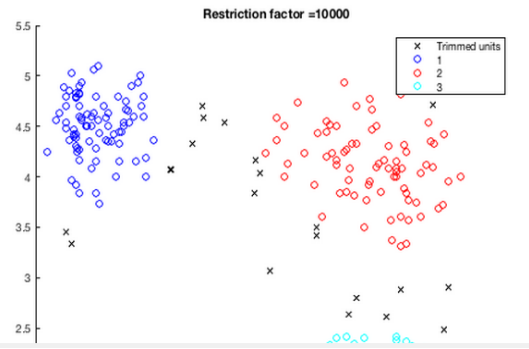
We compare the output using three different values of restriction factor.

```
close all
Y=load('geyser2.txt');
restrfactor=10000;
% nsamp = number of subsamples which will be extracted
nsamp=500;
out=tclust(Y,3,0.1,restrfactor,'nsamp',nsamp,'plots',1);
title(['Restriction factor =' num2str(restrfactor)])
restrfactor=10;
out=tclust(Y,3,0.1,restrfactor,'nsamp',nsamp,'refsteps',10,'plots',1);
title(['Restriction factor =' num2str(restrfactor)])
% trimmed k-means solution restrfactor=1
restrfactor=1;
out=tclust(Y,3,0.1,restrfactor,'nsamp',nsamp,'refsteps',10,'plots',1);
title(['Restriction factor =' num2str(restrfactor) '. Trimmed k-means solution'])
cascade
```

```
Total estimated time to complete tclust:  4.16 seconds
```

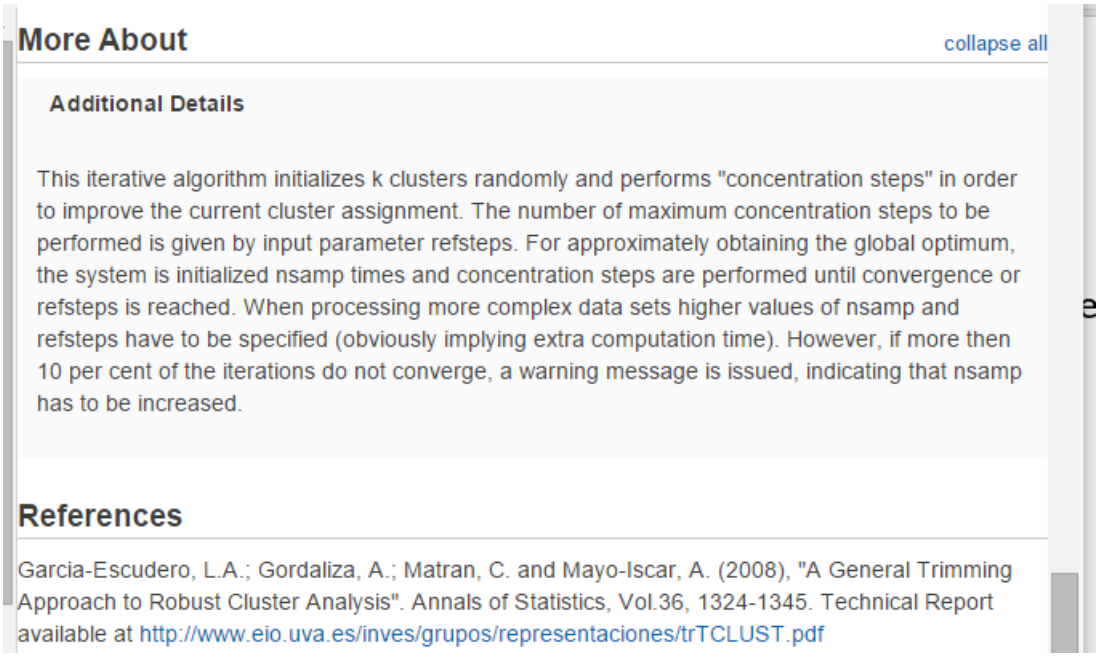






In the `More About` section of our HTML files (see screenshot below), it is possible find the theoretical background which accompanies a particular function.

For example, the screenshot below shows what you get in the case of function `tclust`.



Remark: there is a one to one correspondence between the documentation contained inside the .m file and the corresponding .html file.

The documentation inside the .m file can be easily accessed from the command prompt typing help and the name of the function.

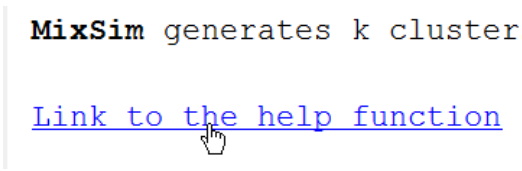
For example, the screenshot below shows what you get if you type in the prompt "`help MixSim`"

```
>> help MixSim
 MixSim generates k clusters in v dimensions with given overlap

 Link to the help function


   Required input arguments:

            k: number of groups (components). Scalar.
               Desired number of groups.
               Data Types - int16|int32|int64|single|double
            v: number of dimensions (variables). Scalar.
               Desired number of variables.
               Data Types - int16|int32|int64|single|double

   Optional input arguments:

     BarOmega : Requested average overlap. Scalar. Value of desired average
                overlap. The default value is ''
                Example - 'BarOmega',0.05
                Data Types - double
     MaxOmega : Requested maximum overlap. Scalar. Value of desired maximum
```

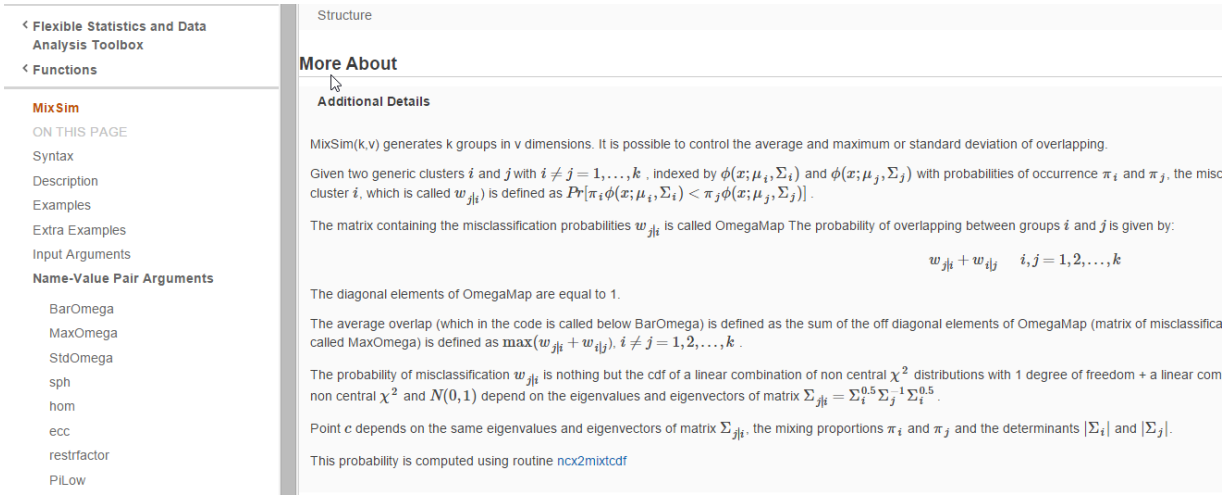
Sometimes inside the .m file (especially in the section "`More About`") we have added a number of formulae in latex language (see screenshot below)

```
More About:

MixSim(k,v) generates k groups in v dimensions. It is possible to
control the average and maximum or standard deviation of overlapping.

Given two generic clusters $i$ and $j$ with $i \ne j =1, ..., k$,
indexed by $\phi(x; \mu_i,\Sigma_i)$ and $\phi(x; \mu_j,\Sigma_j)$ with
probabilities of occurrence $\pi_i$ and $\pi_j$, the misclassification
probability with respect to cluster $i$ (that is conditionally on $x$
belonging to cluster $i$,  which is called  $w_{j|i}$) is defined as
$Pr[ \pi_i \phi(x;\mu_i,\Sigma_i) < \pi_j \phi(x;\mu_j,\Sigma_j)]$.
The matrix containing the misclassification probabilities $w_{j|i}$ is
called OmegaMap
The probability of overlapping between groups $i$ and $j$ is given by:
\[
          w_{j|i} + w_{i|j}     \qquad      i,j=1,2, ..., k
\]
The diagonal elements of OmegaMap are equal to 1.
The average overlap (which in the code is called below BarOmega) is
```

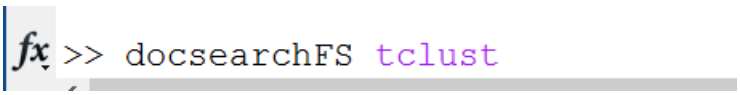
Clearly all these latex formulae will show up correctly (thanks to MathJax technology) in the corresponding HTML help page. For example, in the case of `MixSim` function, in the command window, by clicking on the link “Link to the help function”,



one is redirected to the corresponding HTML documentation page. Here, in the “More About” section it is possible to see the code in proper mathematical style.



Finally, it is worthwhile to remark that it is possible to go directly to the HTML documentation page simply typing `docsearchFS` and the name of the requested function. For example, in the case of `tclust` to reach file tclust.html it is possible to type:



Generally, the output of our functions is a structure, which contains several fields, documented in detail inside the initial part of the .m function. For example, in the case of tclust inside `tclust.m` it is possible to navigate to section `Output` (see screenshot below):

```
%  Output:
%
%         out:   structure which contains the following fields
%
%            out.muopt= k-by-v matrix containing cluster centroid locations.
%                       Robust estimate of final centroids of the groups.
%         out.sigmaopt= v-by-v-by-k array containing estimated constrained
%                       covariance for the k groups.
%            out.idx  = n-by-1 vector containing assignment of each unit to
%                       each of the k groups. Cluster names are integer
%                       numbers from 1 to k. 0 indicates trimmed
%                       observations.
%            out.siz  = matrix of size k-by-3
%                       1st col = sequence from 0 to k
%                       2nd col = number of observations in each cluster
%                       3rd col = percentage of observations in each cluster
%                       Remark: 0 denotes unassigned units
%            out.post = n-by-k matrix containing posterior probabilities
%                       out.post(i,j) contains posterior probabilitiy of unit
%                       i from component (cluster) j. For the trimmed units
%                       posterior probabilities are 0
%          out.MIXMIX = BIC which uses parameters estimated using the
```

In the corresponding HTML file our parser publishFS.m puts all the fields of input and output structure inside a HTML table (see screenshot below):

Output Arguments

out — description
Structure

Expand/Collapse

Structure which contains the following fields

Value	Description
muopt	k-by-v matrix containing cluster centroid locations. Robust estimate of final centroids of the groups.
sigmaopt	v-by-v-by-k array containing estimated constrained covariance for the k groups.
idx	n-by-1 vector containing assignment of each unit to each of the k groups. Cluster names are integer numbers from k. 0 indicates trimmed observations.
siz	matrix of size k-by-3 1st col = sequence from 0 to k 2nd col = number of observations in each cluster 3rd col = percentage of observations in each cluster Remark: 0 denotes unassigned units
post	n-by-k matrix containing posterior probabilities out.post(i,j) contains posterior probabilitiy of unit i from component (cluster) j. For the trimmed units posterior probabilities are 0
MIXMIX	BIC which uses parameters estimated using the mixture loglikelihood and the maximized mixture likelihood as goo of fit measure. Remark: this output is present just if input option mixt is >0
MIXCLA	BIC which uses the classification likelihood based on parameters estimated using the mixture likelihood (In some b this quantity is called ICL) Remark: this output is present just if input option mixt is >0
CLACLA	BIC which uses the classification likelihood based on parameters estimated using the classification likelihood Rema this output is present just if input option mixt is =0
notconver	scalar. Number of subsets without convergence
bs	k-by-1 vector containing the units forming initial subset associated with muopt.
obj	scalar. Value of the objective function which is minimized (value of the best returned solution). If input option mixt >1 the likelihood which is maximized is a mixture likelihood as follows $\prod_{i=1}^{h}\sum_{j=1}^{k}\pi_j\phi(y_i;\theta_j).$

Every subfolder of FSDA contains file `contents.m` (automatically created by our routine `makecontentsfileFS.m`) which contains a series of detailed information about all the .m files of the folder, which have the corresponding HTML documentation. For example, the screenshot referred to the left part of file `contents.m` inside subfolder "utilities" is given below.

```
% UTILITIES
%
% File names, description, category and date last modified
%
%   Name                    - Description
%-------------------------------------------------------------------------------
%   cabc                    - Closes all open figures except the one in foreground (the curre
%   cascade                 - Is a third party function used in FSDA demos and examples
%   clickableMultiLegend    - Hides/shows symbols inside all gplotmatrix subplots (or similar
%   findDir                 - Finds recursively all directories in root
%   findFile                - Finds recursively all files in root
%   isfunction              - Checks if a function exists
%   makecontentsfileFS      - Extends Matlab function makecontentsfile
%   PoolClose               - Closes the pool of MATLAB instances opened with PoolPrepare to
%   PoolPrepare             - Prepares a pool of MATLAB instances for executing code in paral
%   position                - Controls the position of the open figures
%   publishFS               - Enables to create automatic HELP FILES from structured .m funct
%   publishFunctionAlpha    - Enables to create web page which contains the alphabetical list
%   publishFunctionCate     - Enables to create web page which contains the alphabetical list
%   quickselectFS           - Finds the k-th order statistic
%   triu2vec                - Extracts in a vector the linear indexes or the elements on and
%   upperfracpos            - Positions two figures on the upper part of the screen
%   zscoreFS                - Computes (robust) standardized z scores
```

Similarly, inside the main root of FSDA file `contents.m` lists in alphabetical order all files present in all subfolders of FSDA, which have the corresponding HTML page (see screenshot below):

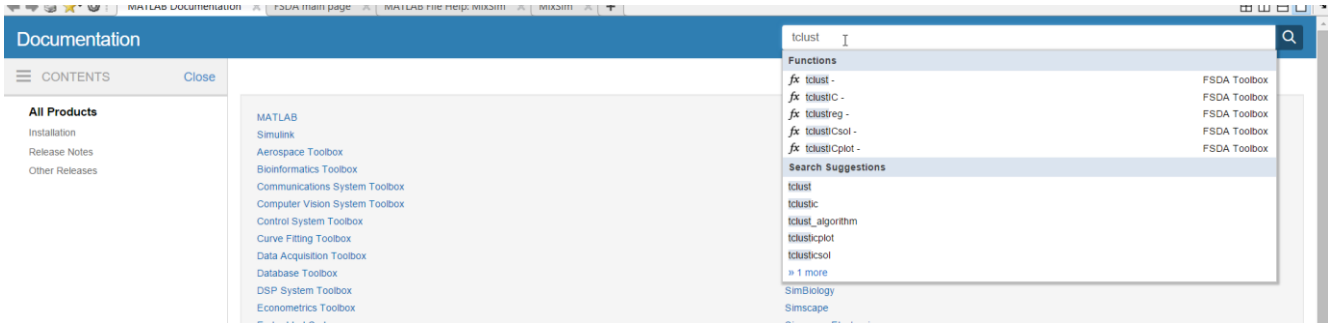
```
% FSDA
%
% File names, description, category and date last modified
%
%   Name                          - Description
%-------------------------------------------------------------------------
%   add2spm                       - Adds objects (personalized clickable multi
%   addt                          - Produces the t test for an additional expl
%   basicPower                    - Computes the basic power transformation
%   bc                            - Returns the Binomial coefficient
%   boxplotb                      - Computes a bivariate boxplot
%   brushFAN                      - Displays a GUI which enables brushing in t
%   brushRES                      - Displays a GUI which enables brushing in r
%   brushROB                      - Displays a GUI which enables brushing in r
%   cabc                          - Closes all open figures except the one in
%   cascade                       - Is a third party function used in FSDA der
%   cdsplot                       - Produces the candlestick plot for robust r
%   clickableMultiLegend          - Hides/shows symbols inside all gplotmatri
%   combsFS                       - Is an iterative algorithm equivalent to th
%   covplot                       - Plots the trajectories of the elements of
%   ellipse                       - Generates an ellipse given mu (location ve
%   fanplot                       - Plots the fan plot for transformation in r
%   findDir                       - Finds recursively all directories in root
```

FSDA html documentation files and MATLAB search engine

Particular attention has been devoted by the FSDA team to have all our HTML files indexed by the old and new search engine of MATLAB. Below we describe what you should get depending on the MATLAB version you have.

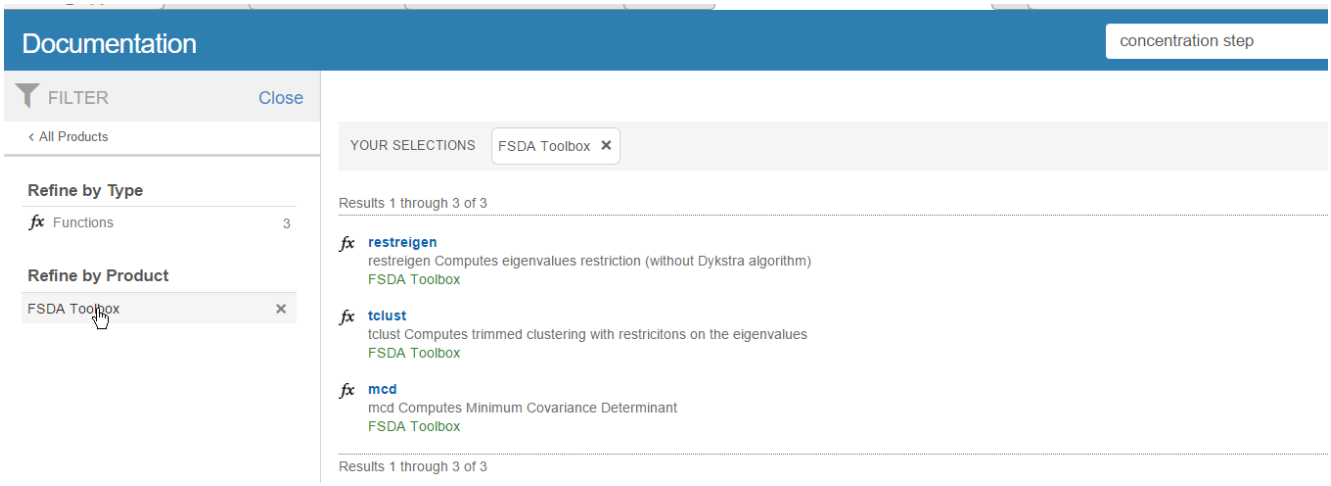
MATLAB 2015a-2016b

If your version of MATLAB is in the range 2015a-2016b, typing inside the engine a word you get also the results from the FSDA toolbox. For example, typing tclust you should automatically have the search suggestion from the drop down menu which automatically appears (see screenshot below)



and you should be brought directly to the tclust documentation page.

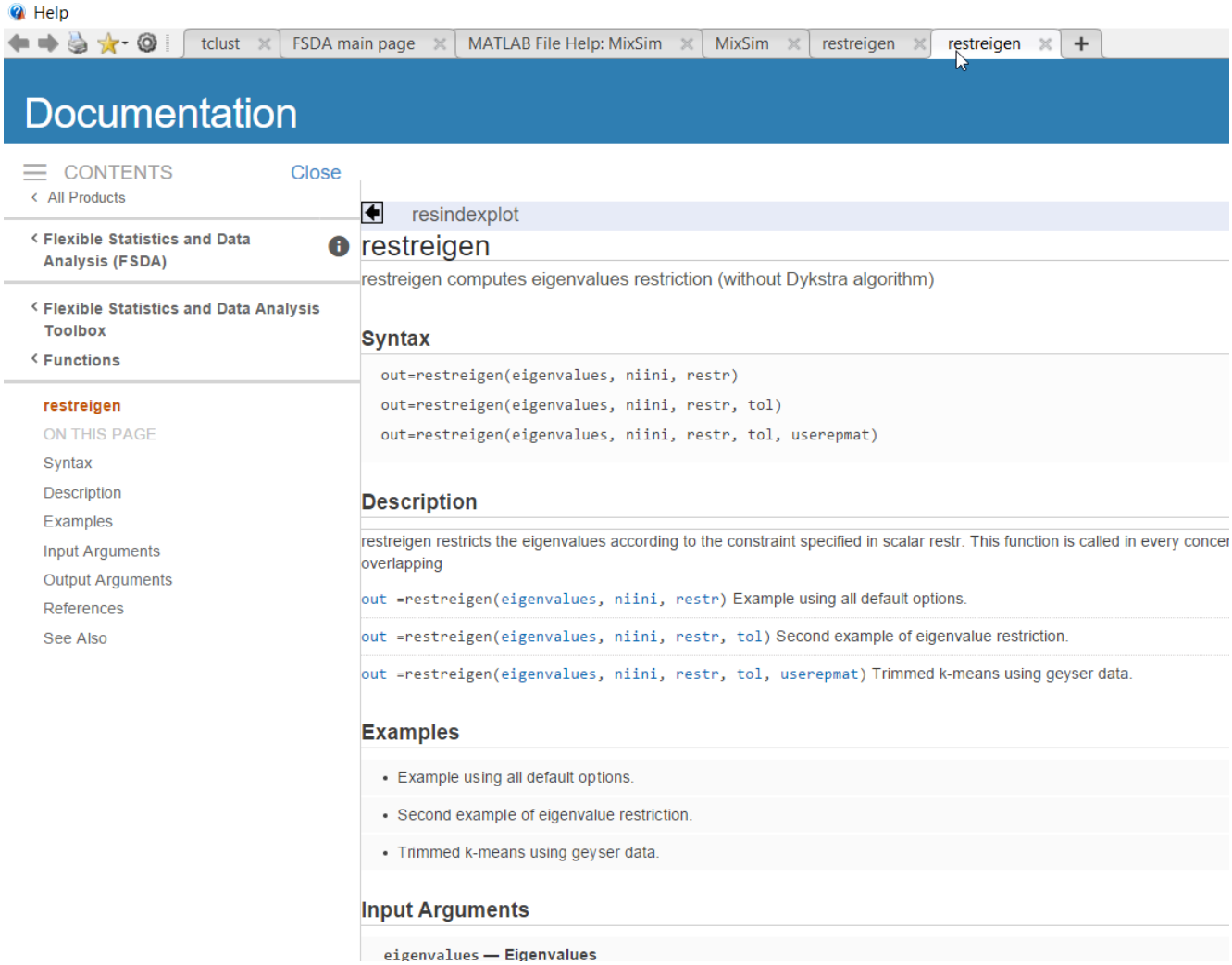
If, for example, you type “concentration step” and you do Refine by Product and select the FSDA toolbox these are the three instances you should get.



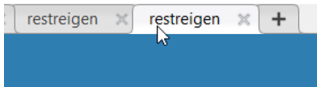
If you put your mouse on the word restreigen you can see from the status bar that the function is located inside (main FSDA folder)/helpfiles/pointersHTML, (screenshot of status bar is given below):

'FSDA/helpfiles/pointersHTML/restreigen.html?searchHighlight=concentration step product:3ptoolbox::fsda_toolbox

Once you click on `restreigen` you can go to page restreigen.html (see screenshot below) which is located inside `docrootFS/FSDA` (see screenshot below)



From the toolstrip on top you will notice that two instances of restreigen have been opened.



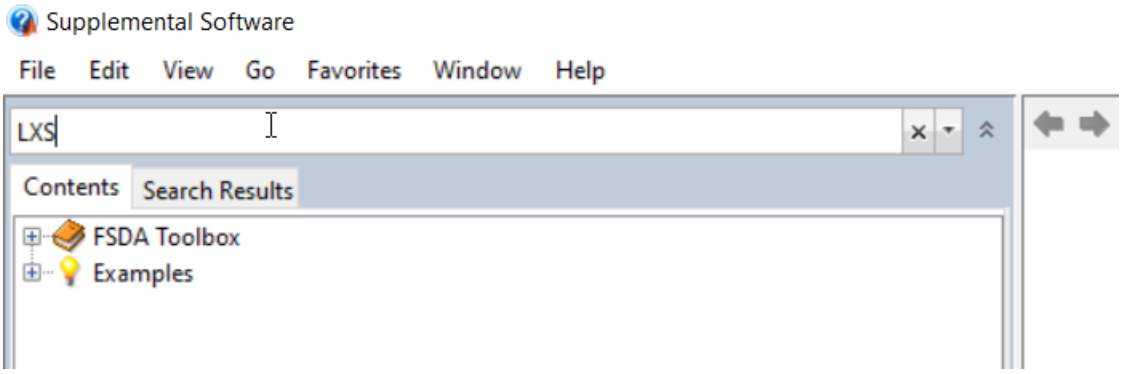
The first on the left is the page which has been indexed by MATLAB search engine which is located inside

(`main FSDA folder`)`/helpfiles/pointersHTML` (see screenshot below):

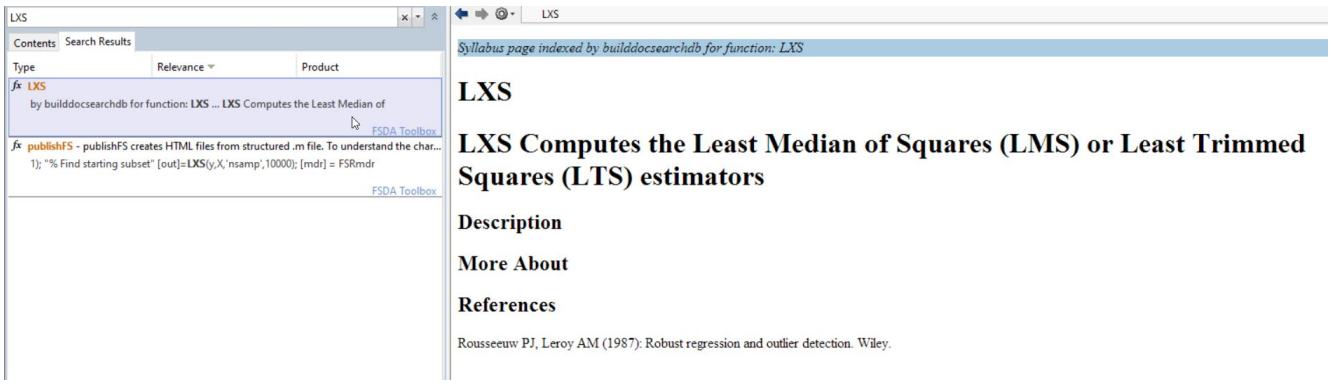
All these syllabus pages have been automatically created by our routine `createFSDAhelp.m`. It was necessary to have the intermediate pages because MATLAB forces these pages to be opened on the iframe on the right. All these syllabus pages contain a redirect to our final HTML pages, which are contained inside `docroot/FSDA`. Files inside `docroot/FSDA` are not forced to be opened on the iframe on the right.

MATLAB 2012b-2014b

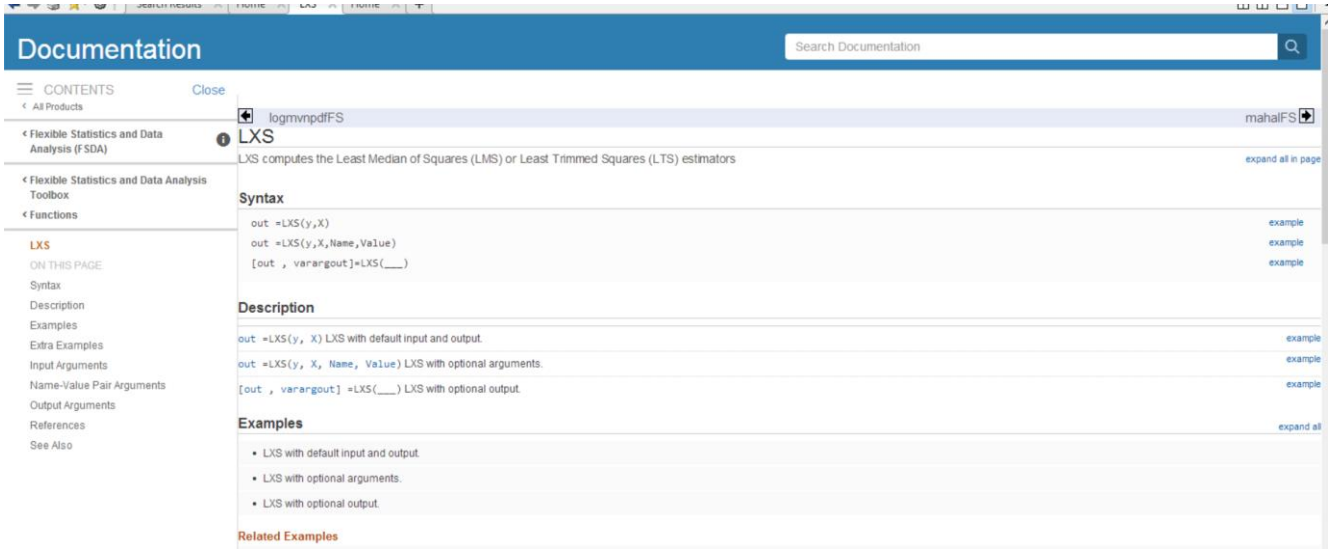
If your version of MATLAB is between 2012b-2014b it is necessary to use the old MATLAB search engine inside supplemental software (see screenshot below)



also, in this case, passing through the syllabus page contained in `(FSDA root)\helpfiles\pointersHTML`

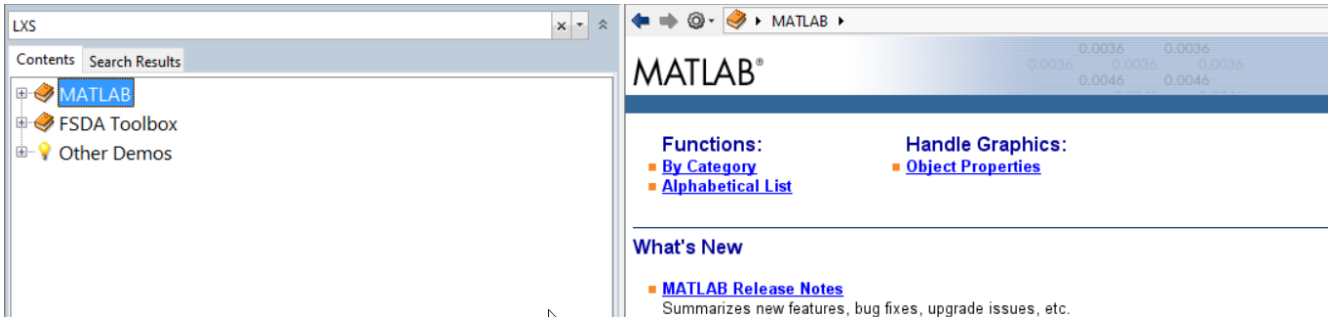


It is possible to automatically reach our page LXS.html contained inside `docroot/FSDA`

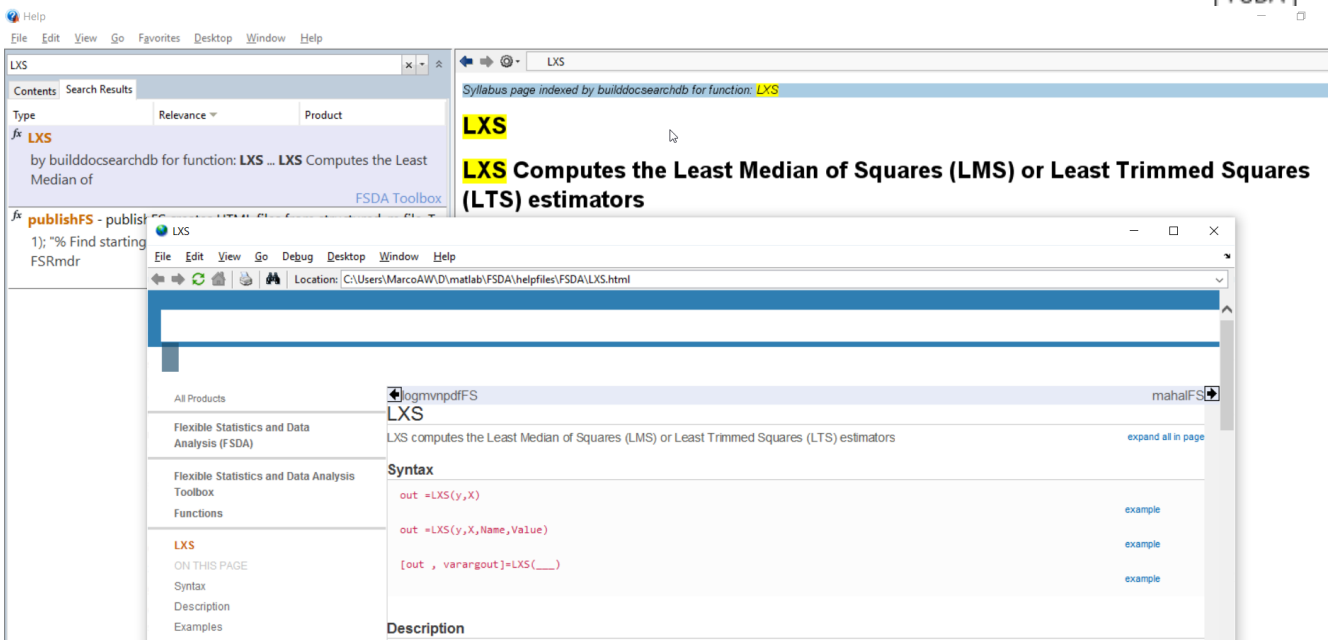


MATLAB <=2012b

In MATLAB older or equal than 2012a, there was no distinction between MATLAB toolboxes and third parties toolboxes (as concerns the documentation), therefore it is possible to search directly from the unique official MATLAB engine. For example searching for LXS



the output of the search is again the syllabus page which automatically redirects to the true HTML page (both are shown in the screenshot below):



Remark 1: note that the old MATLAB browser enables us to see correctly just 95% of the javascripts which characterize the new MATLAB help style.

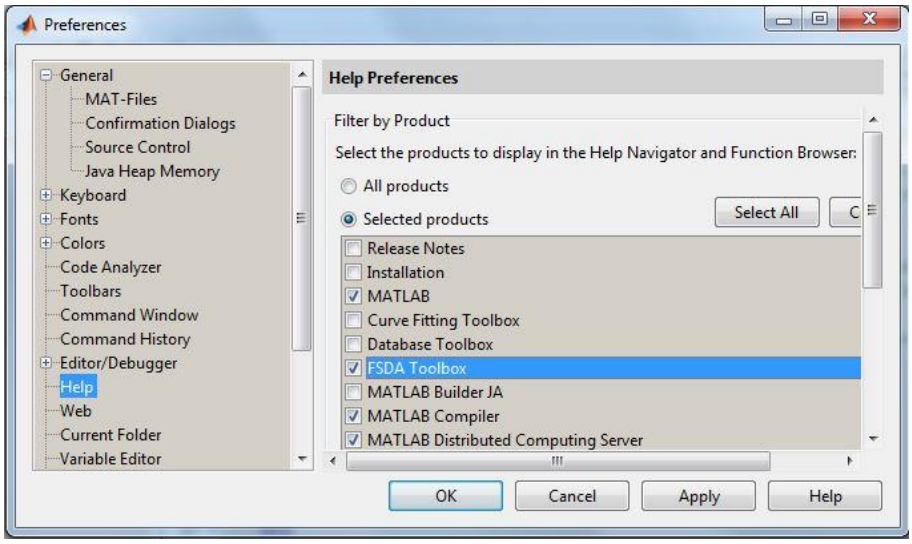
Remark 2: given that in MATLAB versions earlier than 2012b the new engine lucene did not exist, all the "true HTML files" during the installation are not move to folder `docroot/FSDA` but are left inside `(main root of FSDA)/helpfiles/FSDA`.

Remark 3: in order to let the different versions of MATLAB find our documentation files contained inside `(main root of FSDA)/helpfiles/pointersHTML` we have run `buildocsearchFS` using MATLAB 2012a, MATLAB 2014b and MATLAB 2016a. This is the reason why inside `(main root of FSDA)/helpfiles/pointersHTML` there are the 3 subfolders shown in Figure below:

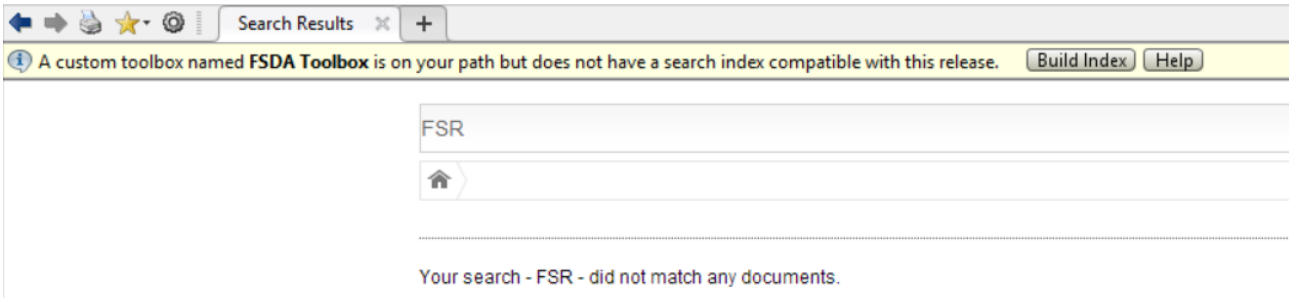
However, if by some obscure reason you cannot find our HTML files using old or new (lucene) engine, it might be necessary to run again `buildocsearchdb`. For example assuming that FSDA main folder is D:\MATLAB\FSDA then it is necessary to run

```
>> buildocsearchdb('D:\MATLAB\FSDA\helpfiles\pointersHTML')
```

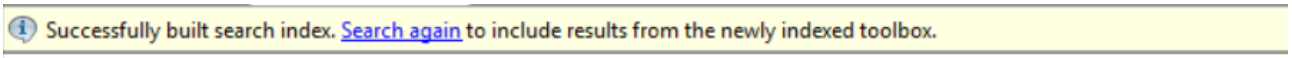
Remark 4: If you are using a release lower than R2012b and you think that the MATLAB Help Browser is not producing proper search results for FSDA functions, check first that in the MATLAB Help Preferences FSDA is selected, as shown in Figure below:



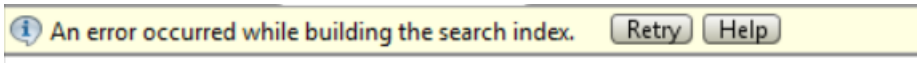
Remark 5: **From MATLAB R2015a, when you search for a given third party function for the first time**, the search results window will display a yellow message warning that a toolbox in the path does not have the proper documentation index file. The window and message produced when attempting to search for documentation about `FSR' function, are shown here:



Only after clicking on the `Build Index` button you should start getting the desired documentation. You will be informed of the successful update of the search database with this message:



If, instead of this message and instead of receiving back the desired results, **you receive an error message** such as this one



again it is likely that you have installed FSDA in a location without proper permissions and, thus, the index building operation (i.e. the `builddocsearchdb` command) could not update the search database. The only solution in this case is to obtain the writing permissions or to change location for FSDA. What you should get if the search is successful, is something like the following:

IF YOU THINK THAT SOMETHING NOT DESCRIBED IN THESE NOTES WENT WRONG

PLEASE DO NOT HESITATE TO SEND AN E-MAIL TO

FSDA@unipr.it[1]

[1] The logo of the toolbox has been designed by Dr. Massimiliano Gusmini.