CatStat

Seismic Catalog and Bulletin Quality Control Analysis MATLAB Package

Table of Contents

License

Unless otherwise noted, this software is in the public domain because it contains materials that originally came from the United States Geological Survey, an agency of the United States Department of Interior. For more information, see the official USGS copyright policy at http://www.usgs.gov/visual-id/credit_usgs.html#copyright

This information is preliminary or provisional and is subject to revision. It is being provided to meet the need for timely best science. The information has not received final approval by the USGS and is provided on the condition that neither the USGS nor the U.S. Government shall be held liable for any damages resulting from the authorized or unauthorized use of the information.

To report any issues, please use GitHub.

Installation

GitHub

Download this code

git clone https://github.com/usgs/CatStat

Creating a local copy of a repository

git clone https://github.com/usgs/CatStat

Seeing the changes you made to a file

git diff FILE

Adding a file to the local copy of a repository

git status
git add FILE
Note: Changing the hardcoded path in the main script will indicate a difference between the local and master branch.

Committing (locally!) the changes you've made to any files

git commit -am "LOG MESSAGE"
git status

Pushing those changes to GitHub

git push origin master

Submit a pull request via the github.com website

Adding changes from the upstream master

Fetch the branches and their respective commits from upstream repository
git fetch upstream
Merge changes into LOCAL master branch, syncing with the upstream without losing local
changes
git merge upstream/master
Review status of repository

> *git status*
>
> Sync local repository to the merged changes
>
> > *git push origin master*

How to re-sync your local copy with the "authoritative" repository

> List the current configured remote repository for your fork
>
> > *git remote -v*
>
> Specify new remote upstream repository to be synced with fork
>
> > *git remote add upstream https://github.com/usgs/CatStat %*
>
> Verify new upstream repository
>
> > *git remote -v*
>
> Then follow directions above for adding changes from the upstream master

> ***For reference:***
>
> > https://help.github.com/articles/configuring-a-remote-for-a-fork/
> >
> > https://help.github.com/articles/syncing-a-fork/

How to fork a new project…

> Set up Git & authentication to GitHub
>
> > Navigate to your fork of the new project repository on GitHub
> >
> > > *git clone URL*
>
> Then follow directions above for syncing with original, upstream repository

> ***For reference:***
>
> > https://help.github.com/articles/fork-a-repo/
> >
> > https://help.github.com/articles/set-up-git/

MATLAB

> In order to run QCreport and QCmulti from any directory, you must add the CatStat directory to your MATLAB path. Adding the following line to your MATLAB *startup.m* file is the simplest way to accomplish this.

> addpath(genpath(***PATH TO CATSTAT***))

Once this is added to your startup.m file, and assuming MATLAB is currently running, simply enter the command '*startup' and* MATLAB will run and read the startup.m script.

> By successfully downloading the GitHub repository and adding the above line to the *startup.m* script, CatStat will be successfully installed. **No additional software or MATLAB packages are needed to run *QCreport* or *QCmulti*.**

> CatStat can be run from the command line with no need to start-up the MATLAB GUI. To do this in bash, add the following aliases to your .bash_profile (or equivalent shell profile file name):

> > alias mkQCmulti='matlab -r "run mkQCmulti.m; quit"'

alias mkQCreport='matlab -r "run mkQCreport.m; quit"'

Catalogue/Bulletin Format Supported

As of 08 November 2016, any CSV formatted file can be used with CatStat. Other delimited files can be used but the user will have to alter the "delimiter" variable within loadcat.m or loadmulti.m. Only one header line may exist, and the following fields must be properly labeled:

ID, OriginTime, Latitude, Longitude, Depth, Mag, Type

Event Type (Type) is optional and if left out will result in all events in a given catalog/bulletin being treated as earthquakes.

Older versions of CatStat only supported 2 types of CSV files. If you still have this version, please upgrade by syncing your local files through Git, or download the newest version.

Single-Catalog QC Report (mkQCReport.m)

The single-catalog QC Report generated by *mkQCReport.m* provides the user with a baseline set of statistics and graphs characterizing and depicting the contents of the catalogue/bulletin. It's purpose is to allow the user to efficiently assess the quality of the data and determine any large error, *i.e.* missing data, systematic catalogue errors, and duplicate postings or entries for the same events. The results are not intended to be a complete investigation of any given catalogue, but as a general, and preliminary look into the completeness and overall quality of data contained in the data set.

Input File (initMkQCreport.dat)

The MATLAB algorithm used to read the contents of the initMkQCreport.dat file relies on line numbers to parse the data. Therefore, order is of the utmost importance.

The following is an example input file (please save input files as initMkQCreport.dat)

% Catalog Data File Name and Path (for completeness, always include full path)
path/to/catalog/dataset/datasetname.csv
% Human readable catalog description
String describing the catalog
% UTC offset [hours]
Integer either + or - depending on timezone
% Time Zone Name
String - Name of timezone for use in plot labels
% Authoritative Region, if none put all
String - See Appendix for abbreviations
% Authoritative Agency, if none put none

String - See [Appendix](#) for abbreviations
% Origin Time in Unix Epoch time?
String - Yes/No
%Directory to Put Report File Into
Example_Report_Directory

How to Run mkQCreport.m

As long as the appropriate steps were taken during [installation](#), mkQCreport can be run from any directory. For example, say I want to run a QC report on the Southern California Catalog and that my catalog is located in /Users/usgs/CI/ as CI.csv. I create a file called "initMkQCreport.dat," and fill out the fields contained in the above example as:

% Catalog Data File Name and Path (for completeness, always include full path)
/Users/usgs/CI/CI.csv
% Human readable catalog description
Southern California Seismic Catalog
% UTC offset [hours]
-8
% Time Zone Name
Pacific Time Zone
% Authoritative Region, if none put all
ci
% Authoritative Agency, if none put none
ci
% Origin Time in Unix Epoch time?
No
%Directory to Put Report File Into
CI_Catalog_Report

Now, within MATLAB, change directories to the folder contained the input file created above (*cd /Users/usgs/CI)* and type *mkQCreport*. This should begin producing a report under the directory specified in the input file. If for some reason you do not see "Using local initMkQCreport.dat file" in the MATLAB command window, then either initMkQCreport.dat isn't in your current working directory, or there was a mistake in your path.

Explanations of the functions and algorithms used can be found in each respective functions header information.

Multiple-Catalog Comparison QC Report (mkQCmulti.m)

The multiple-catalog QC Report generated by *mkQCmulti.m* provides the user with a baseline set of statistics and graphs characterizing and depicting the contents of each catalogue/bulletin, as well as events deemed to be matching, missing, or duplicated between the catalogs/bulletins. It's purpose is to allow the user to efficiently assess the quality of the data and determine any large errors, *i.e.* missing data, systematic catalogue errors, and duplicate postings or entries for the same events. The results are not intended to be a complete investigation of any given catalogue, but as a general, and preliminary look into the completeness and overall quality of data contained in each data set.

Input File (initMkQCmulti.dat)

The MATLAB algorithm used to read the contents of the initMkQCmulti.dat file relies on line numbers to parse the data. Therefore, order is very important.

The following is an example input file (please save input files as initMkQCmulti.dat)

% First catalog data file name
/Users/usgs/CI/CI.csv
% Human readable catalog descriptcity ion
String describing the first catalog
% Second catalog Data file name
/path/to/first/catalog/catalog2.csv
% Human readable catalog description
String describing the second catalog
% Time Window
Time window in seconds for event association (matching time tolerance)
% Distance Range
Distance window in kilometers for event association (matching distance tolerance)
% Authoritative Region; if none put all
String - See Appendix for abbreviations
% Authoritative Agency; if none put none; will turn on authoritative ID check
String - See Appendix for abbreviations
% Set Mag Limit
Lower Magnitude limit to be considered
% Set Mag Difference Max
Magnitude tolerance for event association (set to 9999, if not needed)
% Set Depth difference tolerance
Depth tolerance in kilometers for event association (set to 9999, if not needed)
% Diretory to put report in
Example_Report_Directory

How to Run mkQCmulti.m

As long as the appropriate steps were taken during installation, *mkQCmulti* can be run from any directory. For example, say I want to compare the Southern California Catalog currently in ComCat with a CSV file provided by CI that contains events they tried to upload. The files are located in */Users/usgs/CI/CI.csv* and */Users/usgs/CI/CI_Preupload.csv*.   I create a file called "initMkQCmulti.dat," and fill out the fields contained in the above example as:

> % First catalog data file name
> */Users/usgs/CI/CI.csv*
> % Human readable catalog descriptcity ion
> *CI ComCat*
> % Second catalog Data file name
> */Users/usgs/CI/CI_Preupload.csv*
> % Human readable catalog description
> *CI Preupload*
> % Time Window
> *16*
> % Distance Range
> *100*
> % Authoritative Region; if none put all
> *ci*
> % Authoritative Agency; if none put none; will turn on authoritative ID check
> *ci*
> % Set Mag Limit
> *-10*
> % Set Mag Difference Max
> *0.2*
> % Set Depth difference tolerance
> *10*
> % Diretory to put report in
> *CI_Comparison_Report*

Now, within MATLAB, change directories to the folder contained the input file created above (*cd /Users/usgs/CI)* and type *mkQCmulti*. This should begin producing a report in the specified format (html/xml/pdf etc) under the directory specified on line 12 on the input file. If for some reason you do not see "Using local initMkQCmulti.dat file" in the MATLAB command window, then either initMkQCmulti.dat isn't in your current working directory, or there was a mistake in your path. Because of the way mkQCmulti is currently set-up, the program will not run if it cannot find the input file.

Explanations of the functions and algorithms used can be found in each respective functions header information.

Caveats and Known Bugs

11082016 - No known bugs

Appendix

1. Region Information

    1.1. UW - Northwestern US
    1.2. WY - Wyoming
    1.3. UU - Utah
    1.4. SE - Southeastern US
    1.5. PT - Pacific Tsunami
    1.6. PR - Puerto Rico
    1.7. NN - Nevada
    1.8. NM - New Madrid US Region
    1.9. NC - Northern California
    1.10.   MB - Montana
    1.11.   LD - Lamont-Doherty Authoritative Region
    1.12.   HV - Hawaii Volcano
    1.13.   AT - Atlantic Tsunami
    1.14.   AK - Alaska
    1.15.   CI - Southern California
    1.16.   OK - Oklahoma
    1.17.   KS - Kansas
    1.18.   ALL - Entire Earth
    1.19.   NE - Northeastern US
    1.20.   NE_US - Northeastern US with Canada

2. Authoritative Agency Information and Abbreviations

    2.1. AK - Alaska Earthquake Information Center
    2.2. AT - National Tsunami Warning Center
    2.3. Atlas
    2.4. Choy
    2.5. CI - California Institute of Technology
    2.6. Duputel
    2.7. Gcmt
    2.8. HV - Hawaiian Volcano Observatory
    2.9. ISCGEM