

# CatStat

## Seismic Catalog and Bulletin Quality Control Analysis MATLAB Package

[License](#)

[Installation](#)

[GitHub](#)

[Download this code](#)

[Creating a local copy of a repository](#)

[Seeing the changes you made to a file](#)

[Adding a file to the local copy of a repository](#)

[Committing \(locally!\) the changes you've made to any files](#)

[Pushing those changes to GitHub](#)

[Submit a pull request via the github.com website](#)

[Adding changes from the upstream master](#)

[How to re-sync your local copy with the "authoritative" repository](#)

[How to fork a new project...](#)

[MATLAB](#)

[Catalogue/Bulletin Format Supported](#)

[Catalog Format 1: ComCat CSV](#)

[Catalog Format 2: LibComCat CSV](#)

[Catalog Format 3: ISC-Gem Catalog CSV](#)

[Single-Catalog QC Report \(mkQCReport.m\)](#)

[Input File \(initMkQCReport.dat\)](#)

[How to Run mkQCReport.m](#)

[Detailed Explanation of Algorithms/Methods Used](#)

[mkQCReport.m](#)

[QCReport.m](#)

[loadcat.m](#)

[trimcat.m](#)

[basiccatsum.m](#)

[catalogsize.m](#)

[Multiple-Catalog Comparison QC Report \(mkQCmulti.m\)](#)

[Input File \(initMkQCmulti.dat\)](#)

[Detailed Explanation of Algorithms/Methods Used](#)

[References](#)

[Caveats and Known Bugs](#)

[Appendix](#)

[Region Information](#)



# 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](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.

# 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*.**

# Catalogue/Bulletin Format Supported

As of 1 August 2016, there are only two file formats supported by CatStat, which are explained below.

## Catalog Format 1: ComCat CSV

If the data were obtained using the USGS NEIC API tools or with the ComCat search GUI, the output CSV file is explained [here](#) and contain the following fields:

*time, latitude, longitude, depth, mag, magType, nst, gap, dmin, rms, net, id, updated, place, type, horizontalError, depthError, magError, magNst, status, locationSource, magSource*

The time formats that are currently accepted under this format are:

yyyy-mm-ddTHH:MM:SS.FFFZ  
yyyy-mm-ddTHH:MM:SS.FFF  
yyyy-mm-dd HH:MM:SS.FFFZ  
yyyy-mm-dd HH:MM:SS.FFF  
yyyy-mm-ddTHH:MM:SSZ  
yyyy-mm-ddTHH:MM:SS  
yyyy-mm-dd HH:MM:SSZ  
yyyy-mm-dd HH:MM:SS  
yyyy-mm-dd HH:MM

***Note: Work is currently underway to make the date-time field accept a variety of formats.***

## Catalog Format 2: LibComCat CSV

If the data were obtained using the *getcsv.py* tool in the [LibComCat Python package](#), the output CSV file will currently contain the following fields:

*EventID, Time, Latitude, Longitude, Depth, Mag, EventType*

The time formats that are currently accepted under this format are:

yyyy-mm-ddTHH:MM:SS.FFFZ  
yyyy-mm-ddTHH:MM:SS.FFF  
yyyy-mm-dd HH:MM:SS.FFFZ  
yyyy-mm-dd HH:MM:SS.FFF  
yyyy-mm-ddTHH:MM:SSZ  
yyyy-mm-ddTHH:MM:SS  
yyyy-mm-dd HH:MM:SSZ

yyyy-mm-dd HH:MM:SS  
yyyy-mm-dd HH:MM

***Note: Work is currently underway to make the date-time field accept a variety of formats.***

## Catalog Format 3: ISC-Gem Catalog CSV

If the data were obtained from [ISC](#), or in a similar format, the resulting CSV will contain 59 rows of header information and the following fields:

*Date,lat,lon,smajax,sminax,strike,q,depth,unc,q,mw,unc,q,s,mo,fac,mo\_auth,mpp,mpr,mrr,mrt,mtp,mtt,eventid*

However, only the EventID, date, lat, lon, dep, and Mw columns are used in mkQCReport.m

The time format that are current accepted under this format are:

yyyy-mm-dd HH:MM:SS.FF (native ISC-GEM time format)

***Note: Work is currently underway to make the date-time field accept a variety of formats.***

## 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 very important.

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
% Catalog Format: 1 = ComCat CSV, 2 = libcomcat CSV, 3 = ISC-GEM CSV
Integer specifying catalog format number
```

```
% UTC offset [hours]
Integer either + or - depending on time zone
% Time Zone Name
String - Name of time zone for use in plot labels
%Directory to Put Report File Into
Example_Report_Directory
%File Format for Report (HTML default / PDF) See MATLAB Publish documentation
html
% Show code in report (will be removed in future releases)
false
```

## 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
% Catalog Format: 1 = ComCat CSV, 2 = libcomcat CSV, 3 = ISC-GEM CSV
1
% UTC offset [hours]
-8
% Time Zone Name
Pacific Time Zone
%Directory to Put Report File Into
CI_Catalog_Report
%File Format for Report (HTML default / PDF) See MATLAB Publish documentation
html
% Show code in report (will be removed in future releases)
false
```

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 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 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. Because of the way mkQCreport is currently set-up, the program will run, but will use the default *initMkQCreport.dat* file.



## Detailed Explanation of Algorithms/Methods Used

### mkQCreport.m

This is the parent script responsible for reading the input file and generating the report, through the use of the *publish* command in MATLAB.

### QCreport.m

QCreport.m is the script that ties the algorithms and functions used in generating the statistics and relevant plots for the report. Work is currently underway for each functions to be self-sustaining (i.e. can comment out all but a few functions, and the active functions will still produce output without relying on other “less-important” functions).

### loadcat.m

This function loads catalog/bulletin data from a variety of [formats](#) (ComCat, LibComCat, and ISC-GEM). This loading script requires *mkQCreport.m* to run prior in order to read the input file and gather the needed information i.e., catalog location, format, etc.. If it is desired to run *loadcat.m* without relying on *mkQCreport.m*, the input must be a MATLAB structure with the file location and path saved under *.file*, and the format saved under *.format*.

The output from this function will be a MATLAB data structure with the following fields:

- .name -- name of the catalog (Human readable catalog description from input file)
- .file -- name and path of the catalog file (from input file)
- .data -- Hypocenter information for the catalog (origin time, latitude, longitude, depth, magnitude)
- .id -- Event IDs (rows correspond with rows in .data)
- .evtype -- Event type associated with events in .data

### trimcat.m

If a region other than ‘all’ is selected, this functions will return spatially filtered data using the MATLAB function [inpolygon](#). For region information and abbreviations, please refer to the [appendix](#). Work is currently underway to allow for temporal filtering (an option that is currently working in *mkQCmulti.m*). If a region is not selected (i.e. all), this function is by-passed.

basiccatsum.m

This function provides basic catalog information and statistics, and requires the input to be the data structure created by *loadcat.m*. If the desire is to run this function individually the data structure passed must contain:

- .name - Catalog Name
- .file - Catalog File and Path
- .data - Hypocenter Information (Origin Time, latitude, longitude, depth, magnitude)
- .evtype - Event Type for corresponding event in .data

There is variable output for this function, but it will print out a variety of statistics including catalog time-span, minimum and maximum latitude, longitude, depth, and magnitude. Number of 0 and NaN magnitudes and depths, as well as a count of each event type contained in the catalog.

catalogsize.m

## Multiple-Catalog Comparison QC Report (mkQCmulti.m)

Input File (initMkQCmulti.dat)

Detailed Explanation of Algorithms/Methods Used

References

Caveats and Known Bugs

Appendix

I. Region Information