**Data and codes availability statement**

This document describes the entire RiverLakeBasins analysis workflow, which consists of three main steps: dataset generation, data validation, and plotting. The following describes these three steps, including the data and code used, along with detailed instructions for code usage, data acquisition, and download links. Users are welcome to download these data and code and replicate the work described in this document.

Dataset Production Description

The main dataset production code is located in the lake-river-cat folder. The Python files used are main.py (which generates the tif and shp files for each level, organized into nested folders) and postprocess.py (which merges the shp files in nested folders into a complete hierarchical shp file).

1. Using main.py

First, prepare a configuration file, specifying the location of each parameter. The configuration file (ini) and specific parameters are as follows, using Asia as an example.

Asia.ini Configuration Parameter Table

Project\_root = "./Asia" # Root directory

Basin\_database = "./4\_level.db" # NWEI .shp database address

Alter\_database = "./lake\_alter.db" # Automatically generated lake attribute database address

Lake\_shp = "./Asia\_lakes.shp" # Hydro Lakes .shp file address for the corresponding continent

Minimum\_river\_threshold = 30.0 # Minimum runoff accumulation threshold, change to your needs

Code = 4 # NWEI continent code

Src\_code = 4 # RiverLakeBasins continent code, consistent with NWEI

The data are openly available at https://doi.org/10.5281/zenodo.16888851

To use the code, enter a command prompt window and type:

python main.py [-p process] config level You can run it.

"-p process" indicates the number of CPUs to use for the code, which can be set based on your computer's configuration. "config" refers to the ini configuration file, and "level" indicates the level of the program. You can select from {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}.

2. Using postprocess.py

First, prepare the configuration file (Asia.ini) as above. Then, enter a command prompt and type "Python postprocess.py config level" to run it.

"config" refers to the ini configuration file, and "level" indicates the level of the program. You can select from {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12}.

Data Validation Description

The validation dataset includes RiverLakeBasins and Lake-TopoCat. The data used can be downloaded from https://doi.org/10.5281/zenodo.16888851

The code is stored in compare\_dhp2dic.py. The main functions used are sbatch\_main\_calculate\_upstreamarea\_GNWL, sbatch\_main\_calculate\_upstreamarea\_TopoCat, and merge\_csv\_TopoCat.

sbatch\_main\_calculate\_upstreamarea\_GNWL(venu1, venu2) calculates the upstream area of lakes in RiverLakeBasins. The input parameters are venu1: the directory where the RiverLakeBasins files are stored, and venu2: the output directory for the calculation results. Output files are automatically named, that is, the file in the drawing description.

sbatch\_main\_calculate\_upstreamarea\_TopoCat(venu1, venu2) calculates the upstream area of lakes in Lake-TopoCat. The input parameters are venu1: the directory where the Lake-TopoCat Catchments are stored, and venu2: the output directory for the calculation results. Output files are automatically named, that is, the file in the drawing description.

merge\_csv\_TopoCat(venu2, file2) calculates the CSI verification results of the first two steps. The input parameters are venu2: the directory where the Lake-TopoCat results are stored, and file2: the results of the first step.

The output files are automatically named, i.e., the files in the drawing description, 'dic', 'TopoCatArea.csv', 'Compare'.

Draw Description

1. Draw Figure 6. Run Draw.drawFig1('Global\_statis1.csv') in main.py. Because the Global\_statis1.csv file exceeds Github's upload limit, it cannot be uploaded. The data can be found at 10.5281/zenodo.16809392.

2. Draw Figure 7 in ArcMap. The required data is the RiverLakeBains12-level vector data, available at https://doi.org/10.5281/zenodo.15695045.

3. Draw Figure 8. Run Draw.drawFig2('dic', 'TopoCatArea.csv', 'Compare') in main.py. Because the data files exceed Github's upload limit, they cannot be uploaded. The data can be found at 10.5281/zenodo.16809392.

4. Figure 9. This figure was drawn using Arcmap. The plot file (.mpk) is available at 10.5281/zenodo.16809392 and is titled "Comparison of GNWL and HyBAS Inner Flow Areas."

5. Figure 10. This figure was drawn using Arcmap. The plot file (.mpk) is available at 10.5281/zenodo.16809392 and is titled "Comparison of Lake 16256."

6. The data in Table 2 were obtained from https://doi.org/10.1016/j.scitotenv.2021.145463. For technical details, please refer to this paper, available at 10.5281/zenodo.16809392.

**Data and codes availability statement**

本文档用于描述RiverLakeBasins的整个分析工作流程，主要包含三个主要步骤，数据集的生产，数据验证，绘图。以下将对这三个步骤进行阐述，包含使用的数据和代码，以及代码的使用详解，数据的获取及下载网址。 用户可自行下载这些数据和代码，根据本文档进行工作的重现。

数据集生产描述

数据集生产主代码存在lake-river-cat文件夹下，应用到的python文件有main.py（生产各个层级的tif文件以及shp文件，按照文件夹嵌套的形式进行组织），postprocess.py（将嵌套的文件夹中的shp文件合并成完成的层级shp文件）。

1. main.py使用方法

首先准备配置文件，写明各个参数的地址，配置文件（ini）及具体参数如下，以亚洲为例

Asia.ini 配置参数表

Project\_root = “./Asia” # 根目录地址

Basin\_database = “./4\_level.db” # NWEI的shp数据库地址

Alter\_database = “./lake\_alter.db” # 自动生成的湖泊属性数据库地址

Lake\_shp = “./Asia\_lakes.shp” # 对应的大洲Hydro LAKES的shp文件地址

Minimum\_river\_threshold = 30.0 # 最小汇流累积量阈值，可按照自己的需求更改

Code = 4 # NWEI中对应的每个大洲的编码

Src\_code = 4 # RiverLakeBasins中对应的每个大洲的编码，与NWEI保持一致

数据可在https://doi.org/10.5281/zenodo.16888851开放获取。

使用代码时，需要进入cmd控制窗口，输入

***python main.py [-p process] config level*** 即可运行

其中，-p process是代码使用的cpu数量，用户可根据计算机的配置进行设置，config为ini配置文件，level是划分的层级等级，可在{1,2,3,4,5,6,7,8,9,10,11,12}进行选择。



1. postprocess.py使用方法

首先，准备配置文件，同上（Asia.ini）。接着，进入cmd控制窗口，输入

***Python postprocess.py config level***即可运行

其中，config为ini配置文件，level是划分的层级等级，可在{1,2,3,4,5,6,7,8,9,10,11,12}进行选择。



数据验证描述

数据集验证包含RiverLakeBasins和Lake-TopoCat，使用的数据可在https://doi.org/10.5281/zenodo.16888851下载。

代码存放在Compare\_dhp2dic.py中，主要应用函数为sbatch\_main\_calculate\_upstreamarea\_GNWL、sbatch\_main\_calculate\_upstreamarea\_TopoCat、merge\_csv\_TopoCat。

sbatch\_main\_calculate\_upstreamarea\_GNWL(venu1,venu2)为计算RiverLakeBasins中湖泊上游面积，输入参数为venu1: 存放RiverLakeBains文件的目录地址，venu2:输出计算结果的地址，输出文件均自动命名，即绘图说明中的文件。

sbatch\_main\_calculate\_upstreamarea\_TopoCat(venu1,venu2)为计算Lake-TopoCat中湖泊上游面积，输入参数为venu1: 存放Lake-TopoCar的Catchments的目录地址，venu2:输出计算结果的地址，输出文件均自动命名，即绘图说明中的文件。

merge\_csv\_TopoCat(venu2,file2)为计算前两步计算结果的CSI验证结果，输入参数为venu2为计算的Lake-TopoCat的结果的文件夹目录，file2为第一步的计算结果。

输出文件均自动命名，即绘图说明中的文件，'dic','TopoCatArea.csv','Compare'。

绘图描述

1. 绘制Figure 6. 运行main.py的Draw.drawFig1('Global\_statis1.csv')即可，由于Global\_statis1.csv文件超出Github限制，无法上传，该数据可在10.5281/zenodo.16809392获取；
2. Figure 7在Arcmap软件中绘制，所需数据为RiverLakeBains12层级的矢量，可在[https://doi.org/10.5281/zenodo.15695045](https://doi.org/10.5281/zenodo.15695045" \t "_blank)中获取；
3. 绘制Figure 8. 运行main.py的Draw.drawFig2('dic','TopoCatArea.csv','Compare')，由于各数据文件超出Github限制，无法上传，可在10.5281/zenodo.16809392获取；
4. Figure 9在Arcmap软件中绘制，绘制文件（.mpk）可在10.5281/zenodo.16809392获取，命名为“[GNWL与HyBAS内流区比较](https://zenodo.org/uploads/16809392" \t "_blank) “；
5. Figure 10 Arcmap软件中绘制，绘制文件（.mpk）可在10.5281/zenodo.16809392获取，命名为“[湖泊16256比较](https://zenodo.org/uploads/16809392) “；
6. Table 2 中的数据根据<https://doi.org/10.1016/j.scitotenv.2021.145463获得，具体技术细节请参考此文，可在10.5281/zenodo.16809392>中获取。