# 

# OpenBench用户手册

戴永久教授团队

中山大学

2024年10月4日

## 

## 目录

Table of Contents

[OpenBench用户手册 1](#_Toc179028215)

[目录 2](#_Toc179028216)

[1. 简介 3](#_Toc179028217)

[2. 系统要求 5](#_Toc179028218)

[3. 安装 5](#_Toc179028219)

[4. 快速开始 5](#_Toc179028220)

[5. 配置文件 6](#_Toc179028221)

[5.1 主配置文件（main.nml） 6](#_Toc179028222)

[5.2 模型数据配置文件（sim.nml） 9](#_Toc179028223)

[5.3参考数据配置文件（ref.nml） 11](#_Toc179028224)

[6. 运行评估 15](#_Toc179028225)

[7. 输出解释 15](#_Toc179028226)

[8. 高级用法 16](#_Toc179028227)

[8.1 自定义评估指标 16](#_Toc179028228)

[8.2 并行计算 16](#_Toc179028229)

[8.3 自定义数据处理 16](#_Toc179028230)

[9. 常见问题 16](#_Toc179028231)

[10. 故障排除 16](#_Toc179028232)

# 1. 简介

在地球系统科学和气候变化研究中，陆面模型(Land Surface Models， LSMs)扮演着至关重要的角色。这些模型模拟陆地表面与大气之间的能量、水分和碳循环交换，为我们理解和预测全球气候变化提供了重要工具。然而，随着模型复杂性的增加和应用范围的扩大，如何有效评估和比较不同陆面模型的性能成为了一个日益突出的挑战。基于此，我们开发了The Open Source Land Surface Model Benchmarking System（OpenBench）软件。OpenBench是一个专门设计用于陆面模型评估的开源基准测试系统。OpenBench的目标是为研究人员和模型开发者提供一个标准化、全面且灵活的工具，用于验证模型输出、比较不同模型的性能，并深入分析模型在各种条件下的表现。通过提供一套统一的评估标准和工具，OpenBench旨在促进陆面模型的持续改进和科学社区的协作。

OpenBench的设计秉承着开放、标准、灵活、全面和高效的核心理念。开放性体现在OpenBench作为一个开源项目，鼓励科学社区的广泛参与和贡献，致力于提高评估方法的透明度和可复现性。标准化方面，OpenBench通过提供统一的评估指标和数据处理流程，确保了不同研究之间结果的可比性。灵活性则体现在系统设计允许用户根据特定需求自定义评估流程，并支持各种类型的陆面模型和数据格式。全面性方面，OpenBench不仅关注单一变量的表现，还提供了多变量、多尺度的综合评估能力。最后，效率方面，OpenBench通过并行计算和优化的数据处理流程，能够高效处理大规模数据集。

OpenBench的功能强大且全面，为陆面模型评估提供了一站式解决方案。它允许同时处理多个模型的输出，方便研究人员直接比较不同模型在相同条件下的表现，这对于模型比较、集成和不确定性分析等研究至关重要。OpenBench支持对众多陆面过程变量进行评估，涵盖了生态系统碳循环、水文循环、能量平衡和生物物理参数等方面，并为每个变量提供了一套全面的评估指标，例如总初级生产力、蒸散发、土壤湿度等，全方位评估模型性能。不仅如此，OpenBench还提供了基本统计指标和高级评分方法，例如Kling-Gupta效率、Nash-Sutcliffe效率等，并支持时间序列分析和空间模式评估，能够深入分析模型在不同时空尺度上的表现。OpenBench支持在小时到年际的时间尺度、站点到全球的空间尺度以及不同生态系统类型上进行评估，使用户能够全面了解模型在各种条件下的表现。为了确保评估结果的可靠性，OpenBench集成了自动数据格式转换、时空插值、缺失数据处理和异常值检测等功能，保证了输入数据的质量和一致性。此外，OpenBench还提供了丰富的可视化工具，包括时间序列图、空间分布图、Taylor图等，不仅有助于结果解释，还能生成高质量的图表用于报告和发表。最后，OpenBench实现了并行计算，用户可以根据硬件资源配置计算核心数，从而高效处理大规模数据集，显著提高处理速度。

OpenBench采用模块化设计，其核心组件包括配置管理模块、数据处理模块、评估引擎、比较分析模块、统计分析模块、可视化模块和报告生成器。配置管理模块负责读取和解析用户配置文件，设置评估参数。数据处理模块处理模型输出和参考数据，进行格式转换、对齐和质量控制。评估引擎是实现各种评估指标和评分方法的核心模块。比较分析模块用于多模型比较和集成分析，统计分析模块提供高级统计分析功能。可视化模块生成各种图表和可视化结果，报告生成器汇总结果并生成综合评估报告。OpenBench的工作流程高度自动化，用户只需准备配置文件和输入数据，即可获得全面的评估结果。系统首先读取配置，加载并预处理数据，然后执行指定的评估指标计算。如果开启比较功能，系统会进行多模型比较分析。接下来，系统会根据配置执行统计分析，最后生成可视化结果和评估报告。

OpenBench在陆面模型研究和应用中用途广泛。它可以帮助模型开发者验证新开发的模型模块，评估参数化方案的改进，识别模型的优势和不足，从而推动模型的改进和完善。OpenBench也适用于多模型比较研究，例如气候变化影响评估。它能够确保结果的可比性，揭示模型间的差异，并支持不同类型的比较分析，为研究人员提供了一个理想的工具。此外，OpenBench还可以集成到模型校准和参数优化流程中，为优化算法提供目标函数，评估不同参数设置下的模型性能，并分析参数敏感性，从而帮助研究人员找到最佳的模型参数。在数据同化和模型-数据融合应用中，OpenBench可以评估同化前后的模型性能，为同化算法提供误差统计信息，并验证同化结果的时空一致性，最终提高数据同化的效果。OpenBench也能够用于气候变化影响评估，它可以评估模型在历史期的表现，分析未来情景下的模型一致性，并量化预测的不确定性，为气候变化影响评估提供可靠的依据。总而言之，OpenBench为陆面模型研究提供了全面的评估和分析工具，促进了模型的发展和应用，并为天气和气候等相关研究提供了有力支持。

# 2. 系统要求

* Python 3.10+
* 依赖库: netcdf4，xarray， numpy， pandas，matplotlib， dask， scipy，bottleneck， flox，cartopy，matplotlib以及joblib
* 足够的存储空间用于处理和存储大型数据集

# 3. 安装

1. 克隆 OpenBench 仓库:

* git clone git@github.com:zhongwangwei/OpenBench.git

1. 进入 OpenBench 目录:

* cd OpenBench

1. 安装依赖:

* 使用conda安装：conda install -c conda-forge dask netCDF4 bottleneck flox xarray scipy cartopy matplotlib joblib pandas
* 使用pip安装：pip install -r requirements.txt 其中requirements.txt 文件列出了项目所需的所有Python包及其版本。这个文件位于项目的根目录下。

# 4. 快速开始

1. 准备您的模型输出数据和参考数据。
2. 创建或修改配置文件 (main.nml， sim.nml， ref.nml)。
3. 运行 OpenBench:

* python script/openbench.py nml/main.nml

1. 查看输出目录中的结果。

# 5. 配置文件

OpenBench 使用三个主要的配置文件，主配置文件（main.nml），模型数据配置文件（sim.nml）和参考数据配置文件（ref.nml），这三个文件的命名可以随意。

## 5.1 主配置文件（main.nml）

主配置文件（main.nml ）采用Fortran namelist格式，定义了整个评估过程的参数,包括基本设置、评估项目、评估指标、比较方法和统计分析等。示例:

&general  
 basename = "test\_run"  
 basedir = "/path/to/output"  
 compare\_tim\_res = "Month"  
 compare\_tzone = "UTC"  
 compare\_grid\_res = 0.5  
 syear = 2002  
 eyear = 2003  
 reference\_nml = "./nml/ref.nml"  
 simulation\_nml = "./nml/sim.nml"  
 statistics\_nml = "./nml/stats.nml"  
 num\_cores = 16  
 evaluation = True  
 comparison = True  
 statistics = False  
/  
  
&evaluation\_items  
 Gross\_Primary\_Productivity = True  
 Ecosystem\_Respiration = False  
 Net\_Ecosystem\_Exchange = False  
 ...  
/  
  
&metrics  
 percent\_bias = True  
 absolute\_percent\_bias = False  
 bias = False  
 ...  
/  
  
&scores  
 nBiasScore = True  
 nRMSEScore = True  
 nPhaseScore = True  
 ...  
/  
  
&comparsions  
 IGBP\_groupby = True  
 PFT\_groupby = True  
 HeatMap = True  
 ...  
/  
  
&statistics  
 Mann\_Kendall\_Trend\_Test = False  
 Correlation = False  
 Standard\_Deviation = False  
 ...  
/

如上所示，该文件包含多个 namelist 块,每个块都以 & 开始,以 / 结束。主要的块包括：

* &general: 一般设置
* &evaluation\_items: 评估项目
* &metrics: 评估指标
* &scores: 评分方法
* &comparsions: 比较分析
* &statistics: 统计分析

其中一般设置（&general）块包含了评估的基本设置：

* basename: 评估运行的名称
* basedir: 输出目录的路径
* compare\_tim\_res: 比较的时间分辨率 (e.g., "Month", "Day")
* compare\_tzone: 时区设置
* compare\_grid\_res: 空间网格分辨率（度）
* syear, eyear: 评估的起始和结束年份
* max\_lat, min\_lat, max\_lon, min\_lon: 评估区域的地理范围
* min\_year: 最小评估年限
* reference\_nml, simulation\_nml, statistics\_nml: 其他配置文件的路径
* num\_cores: 并行计算使用的核心数
* evaluation, comparison, statistics: 开启/关闭不同的评估模块

&evaluation\_items 块定义了目前可以进行评估的变量，以及是否进行评估的开关。每个变量可以设置为 True (评估) 或 False (不评估)， 同理，&metrics 块定义了可以使用的评估指标及其开关，而&scores 块定义评分指标及其开关。&comparsions 块定义多模式比较时所使用的比较分析方法，&statistics 块定义了统计分析的方法。

## 5.2 模型数据配置文件（sim.nml）

模型数据配置文件（sim.nml），定义了模型输出数据的位置、格式、变量名称等信息,是系统正确读取和处理模型数据的关键。同主配置文件一致，文件采用Fortran namelist格式,包含多个块:

* &general: 通用设置,定义评估项目的数据源
* &def\_nml: 定义具体模型的配置文件路径
* 每个模型的具体配置块(例如 &CoLM2024, &CLM5 等)

其中**&general 块**定义了每个评估项目的模型数据源：如

&general

Gross\_Primary\_Productivity\_sim\_source = CoLM2024, CLM5

Evapotranspiration\_sim\_source = CoLM2024, CLM5

Canopy\_Transpiration\_sim\_source = CoLM2024, CLM5

Total\_Runoff\_sim\_source = CoLM2024, CLM5

Latent\_Heat\_sim\_source = CoLM2024, CLM5

Sensible\_Heat\_sim\_source = CoLM2024, CLM5

...

/

其中每一行定义一个评估项目的数据源 ，同时可以指定多个模型,用逗号分隔，值得注意的是模型名称应与后面定义的具体配置块（&def\_nml 块）名称一致。&def\_nml 块定义了每个模型的具体配置文件路径，如：

&def\_nml

CLM5 = ./nml/user/CoLMvsCLM5/IHistClm50BgcCropGSWP\_CWD-1901-2014.nml

CoLM2024 = ./nml/user/CoLM20X4vsCLM5/CoLM2024.nml

CoLM2014 = ./nml/user/CoLM20X4vsCLM5/CoLM2014.nml

/

&def\_nml里面的CLM5，CoLM2024和CoLM2014是模式运行的实例，可以是任意名称，只要和&general里面的名称保持一致即可。每个实例都有自己的配置块,定义该模型输出数据的详细信息。以CoLM2024为例:

&CoLM2024

model\_namelist = ./nml/Mod\_variables\_defination/CoLM.nml

timezone = 0.0

data\_type = grid

data\_groupby = month

dir = /path/to/CoLM2024/output/

fulllist =

tim\_res = month

grid\_res = 0.5

suffix =

prefix = Global\_Grid\_2x2\_PFT\_VG\_BGC-Hist1850-2021\_hist\_

syear = 2002

eyear = 2003

/

其中包括

* model\_namelist: 模型变量定义文件路径
* timezone: 数据的时区
* data\_type: 数据类型(grid或stn)
* data\_groupby: 数据的时间分组方式
* dir: 数据文件目录
* tim\_res: 时间分辨率
* grid\_res: 空间分辨率
* suffix/prefix: 文件名后缀/前缀
* syear/eyear: 起始/结束年份

要配置多个模型输出,需要修改模型数据配置文件的以下内容:

* 在&general块中为每个评估项目列出所有模型
* 在&def\_nml块中为每个模型定义配置文件路径
* 如果模型变量定义文件路径（默认 ./OpenBench/nml/Mod\_variables\_defination）中不存在对应的模型变量定义文件,则需要创建一个自定义的变量定义文件，命名可以随意。

## 5.3参考数据配置文件（ref.nml）

参考数据配置文件（ref.nml），定义了用于评估模型性能的参考数据的来源、格式、变量名称等信息。这些参考数据可能包括实地观测、卫星数据、再分析数据等。同样，文件采用Fortran namelist格式,包含多个块:

* &general: 通用设置,定义每个评估项目的参考数据源
* &def\_nml: 定义具体参考数据集的配置文件路径
* 各个参考数据集的具体配置块(例如 &FLUXCOM, &ILAMB\_monthly 等)

其中&general 块这个块定义了每个评估项目的参考数据源，如

&general

Sensible\_Heat\_ref\_source = FLUXCOM

Latent\_Heat\_ref\_source = FLUXCOM

Ground\_Heat\_ref\_source = ILAMB\_monthly

Net\_Radiation\_ref\_source = FLUXCOM

Gross\_Primary\_Productivity\_ref\_source = FLUXCOM-X-BASE\_monthly

Net\_Ecosystem\_Exchange\_ref\_source = ILAMB\_monthly

Ecosystem\_Respiration\_ref\_source = ILAMB\_monthly

...

/

其中每一行定义一个评估项目的参考数据源，可以指定多个数据源,用逗号分隔，数据源名称应与后面定义的具体配置块（&def\_nml）名称一致。&def\_nml 块定义了每个参考数据集的具体配置文件路径，如：

&def\_nml

FLUXCOM = ./nml/Ref\_variables\_defination/Composite/FLUXCOM.nml

FLUXCOM-X-BASE\_monthly = ./nml/Ref\_variables\_defination/Composite/FLUXCOM-X-BASE\_monthly.nml

ILAMB\_monthly = ./nml/Ref\_variables\_defination/Composite/ILAMB\_monthly.nml

GLEAM4.1a\_monthly = ./nml/Ref\_variables\_defination/Composite/GLEAM4.1a\_monthly.nml

ERA5LAND = ./nml/Ref\_variables\_defination/Composite/ERA5Land\_daily.nml

...

/

参考数据集都有自己的配置块定义该数据集的详细信息，默认保存在/OpenBench/nml/Ref\_variables\_defination文件夹内。以FLUXCOM为例:

&FLUXCOM

root\_dir = /path/to/FLUXCOM/data/

timezone = 0.0

data\_type = grid

data\_groupby = Single

tim\_res = Month

grid\_res = 0.5

/

&Sensible\_Heat

sub\_dir = Sensible\_Heat/FLUXCOM

varname = sh

varunit = w m-2

prefix = sh

suffix =

syear = 1980

eyear = 2014

/

&Latent\_Heat

sub\_dir = Latent\_Heat/FLUXCOM

varname = le

varunit = w m-2

prefix = le

suffix =

syear = 1980

eyear = 2014

/

其中：

root\_dir: 数据文件的根目录

timezone: 数据的时区

data\_type: 数据类型(grid或stn)

data\_groupby: 数据的时间分组方式

tim\_res: 时间分辨率

grid\_res: 空间分辨率

每个评估变量有自己的小块，定义以下内容:

* sub\_dir: 子目录（如有）
* varname: 变量名
* varunit: 变量单位
* prefix/suffix: 文件名前缀/后缀
* syear/eyear: 起始/结束年份

如果需要加入新的自定义参考数据集,则需要创建一个自定义的变量定义文件，命名可以随意，内容可以参考上述FLUXCOM变量定义文件。

# 6. 运行评估

运行 OpenBench:

python script/openbench.py nml/main.nml

系统将按照配置文件中的设置执行以下步骤:

1. 数据处理

2. 验证

3. 比较

4. 统计分析

# 7. 输出解释

评估结果将保存在 basedir/basename/output/ 目录下，包括:

* metrics/: 包含各种评估指标的结果
* scores/: 包含评分结果
* comparisons/: 包含比较分析结果
* statistics/: 包含统计分析结果

每个子目录中都有相应的图表和数据文件。

# 8. 高级用法

## 8.1 自定义评估指标

用户可以通过修改 Mod\_Metrics.py 和 Mod\_Scores.py 来添加新的评估指标和评分方法。

## 8.2 并行计算

通过在 main.nml 中设置 num\_cores，可以启用并行计算以提高性能。

## 8.3 自定义数据处理

如需添加对新数据格式的支持，可以修改 Mod\_DatasetProcessing.py。

# 9. 常见问题

Q: 如何添加新的评估变量?

A: 在 main.nml 的 &evaluation\_items 部分添加新变量，并确保在 sim.nml 和 ref.nml 中提供相应的数据路径。

Q: 如何处理不同时间分辨率的数据?

A: OpenBench 会自动将数据重采样到 compare\_tim\_res 指定的时间分辨率。确保在配置文件中正确设置了原始数据的时间分辨率。

# 10. 故障排除

* 如果遇到内存错误，尝试减少 num\_cores 或使用更小的数据集进行测试。
* 确保所有输入数据路径在配置文件中正确设置。
* 检查 Python 版本和所有依赖库是否已正确安装。