
cnmaps

发布 0.2.1

Clarmy

2022 年 04 月 05 日

1	cnmaps 使用指南	3
1.1	安装	3
1.2	快速开始	3
1.3	使用案例	4
1.3.1	绘制各省地图	4
1.3.2	合并省界	6
1.3.3	绘制青藏高原	7
1.3.4	根据地图边界裁剪填色等值线	8
1.3.5	根据边界裁减填色网格图	9
1.3.6	调整图片边界位置	10
1.3.7	剪切等值线图	11
1.3.8	对 label 的裁减	12
1.3.9	变换投影	13
1.4	资料引用	15
1.5	版本日志	15
1.5.1	0.2.1	15
1.5.2	0.2.0	16
1.5.3	0.1.11	16
1.5.4	0.1.10	16



cnmaps 是一个致力于让中国地图的获取和使用更丝滑的 python 扩展包。

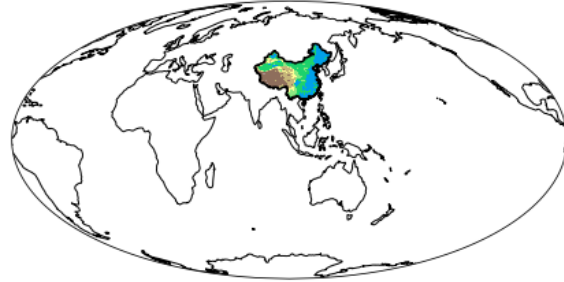
当前版本具有以下几个主要功能：

1. 自带合规地图边界，数据源来自于高德等测绘机构，让你无需再额外寻找地图边界文件。
2. 支持地图边界之间的加减、交并集等常规操作，让你可以自由地组合想要的地图形状。
3. 具有易于使用的地图裁剪功能，且裁剪效果好，平滑无锯齿。
4. 与 cartopy 集成，可以自动转换地图边界的投影。

Mercator



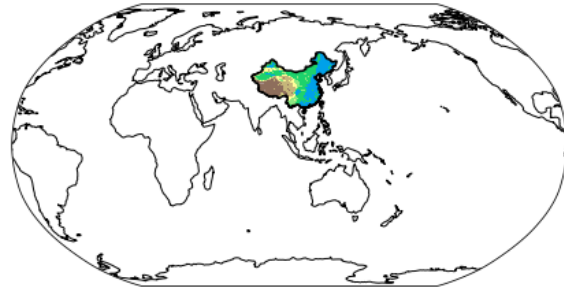
Mollweide



Orthographic



Robinson



1.1 安装

cnmaps 依赖于 `cartopy>=0.19.0`，因此在安装 cnmaps 之前请确保 cartopy 已安装，[cartopy 的安装方法](#)。在完成 cartopy 的安装以后，你可以使用 pip 来安装 cnmaps: `$ pip install cnmaps==0.2.1`

警告： 由于 cnmaps 目前为探索和实验阶段，有些功能可能会随着版本的更新而发生巨大的变化，为了避免由于版本更新而导致代码不可用，请在安装时指定版本号。本文档是以 `cnmaps==0.2.1` 版本进行说明的，其他版本可能并不适用。

1.2 快速开始

用最简单直接的方式，来绘制你的第一张中国地图。

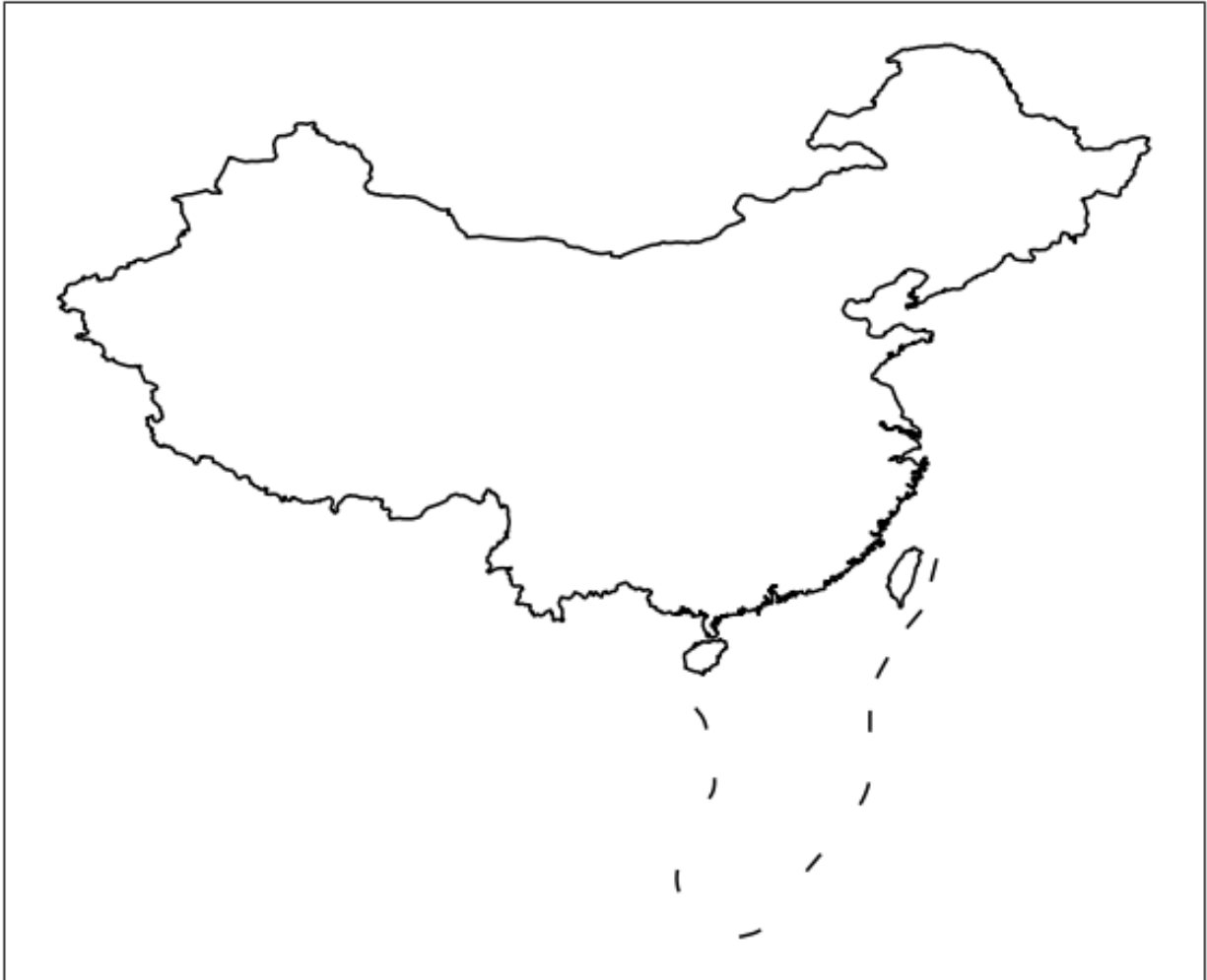
```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map

fig = plt.figure(figsize=(10,10))
ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
draw_map(get_map('中国'), color='k')
```

(下页继续)

(续上页)

```
draw_map(get_map('南海'), color='k')  
  
plt.show()
```



1.3 使用案例

1.3.1 绘制各省地图

分别绘制中国及河南省的边界。

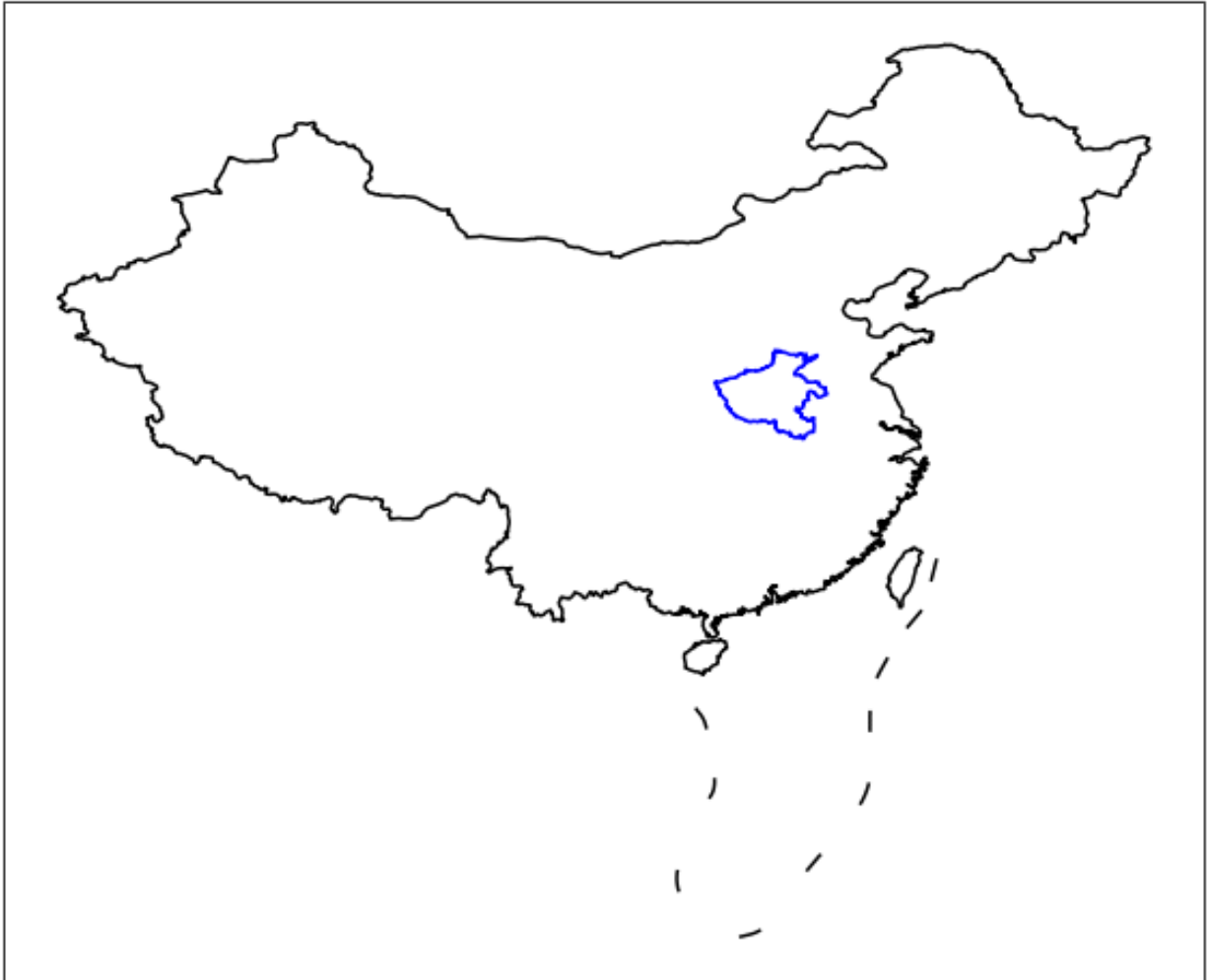
```
import cartopy.crs as ccrs  
import matplotlib.pyplot as plt  
from cnmaps import get_map, draw_map
```

(下页继续)

(续上页)

```
fig = plt.figure(figsize=(10,10))
ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
draw_map(get_map('中国'), color='k')
draw_map(get_map('南海'), color='k')
draw_map(get_map('河南'), color='b')

plt.show()
```



1.3.2 合并省界

将多个省（特区/直辖市）合并起来，我们用很简单的方式来可以绘制一张京津冀的轮廓图。

```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map

jingjinji = get_map('北京') + get_map('天津') + get_map('河北')

fig = plt.figure(figsize=(10,10))
ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
draw_map(jingjinji, color='k')

plt.show()
```



1.3.3 绘制青藏高原

cnmaps 还内置了青藏高原的边界，可以直接调取使用。

```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map

fig = plt.figure(figsize=(10,10))
```

(下页继续)

```
ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
draw_map(get_map('青藏高原', map_set='geography'), color='k')

plt.show()
```



1.3.4 根据地图边界裁剪填色等值线

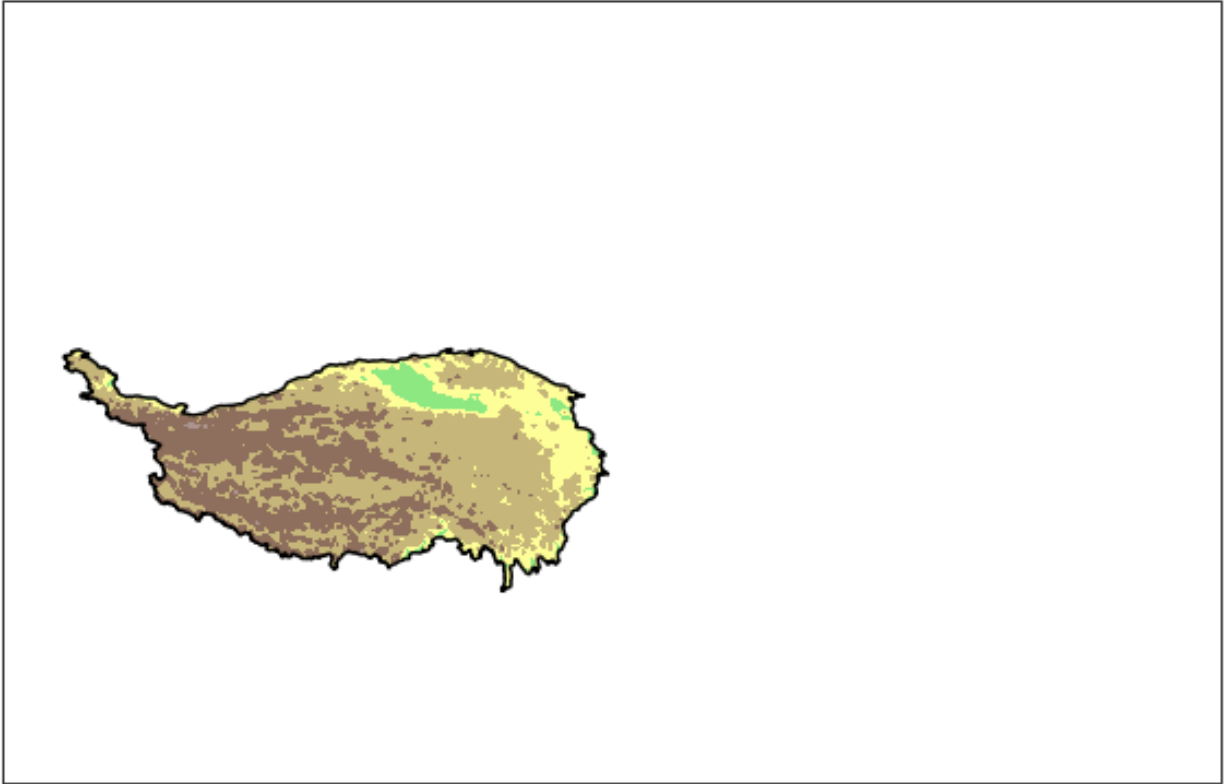
cnmaps 可以利用地图边界对等值线图进行裁减，只需要一个 `clip_contours_by_map` 函数即可。

```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_contours_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()
fig = plt.figure(figsize=(10,10))

tp = get_map('青藏高原', map_set='geography')

ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
cs = ax.contourf(lons, lats, dem, cmap=plt.cm.terrain)
clip_contours_by_map(cs, tp)
draw_map(tp, color='k')
```



1.3.5 根据边界裁减填色网格图

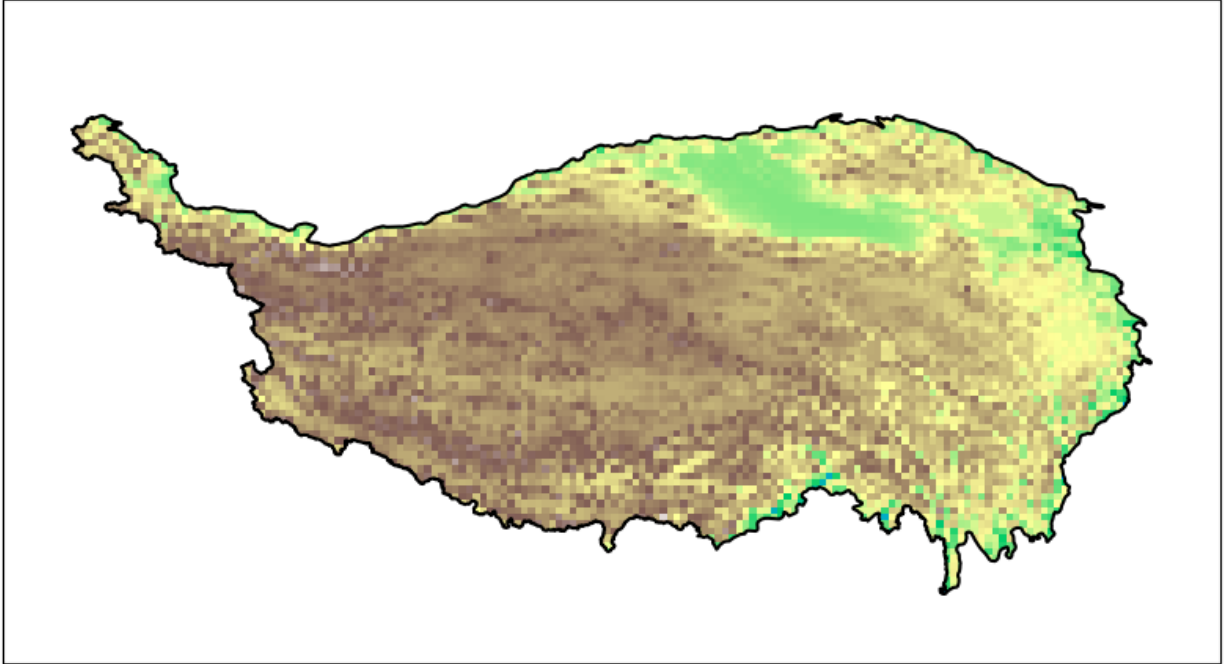
cnmaps 也可以对网格图进行裁减，使用 `clip_pcolormesh_by_map` 函数即可。

```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_pcolormesh_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()
fig = plt.figure(figsize=(10, 10))

tp = get_map('青藏高原', map_set='geography')

ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
mesh = ax.pcolormesh(lons, lats, dem, cmap=plt.cm.terrain)
clip_pcolormesh_by_map(mesh, tp)
draw_map(tp, color='k')
ax.set_extent(tp.get_extent())
```



1.3.6 调整图片边界位置

我们可以利用 `get_extent` 方法获取不同缩放等级的边界，例如下图，我们用 12 个不同等级的缩放来绘制青藏高原的海拔高度图

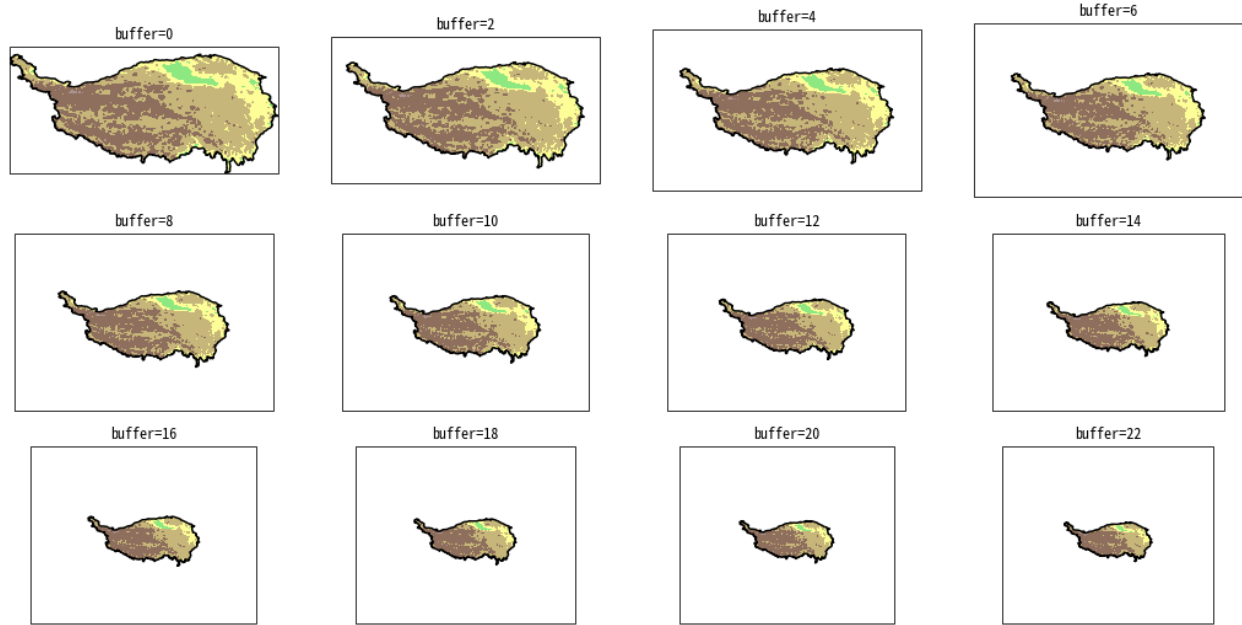
```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_contours_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()
fig = plt.figure(figsize=(12,6))
fig.tight_layout()

tp = get_map('青藏高原', map_set='geography')

for i in range(12):
    ax = fig.add_subplot(3,4,i+1, projection=ccrs.PlateCarree())
    cs = ax.contourf(lons, lats, dem, cmap=plt.cm.terrain)
    clip_contours_by_map(cs, tp)
    draw_map(tp, color='k')
    ax.set_extent(tp.get_extent(buffer=i*2))
    plt.title(f'buffer={i*2}')

plt.show()
```



1.3.7 剪切等值线图

除了填色等值线，非填色的等值线也可以直接用 `clip_contours_by_map` 进行剪切。

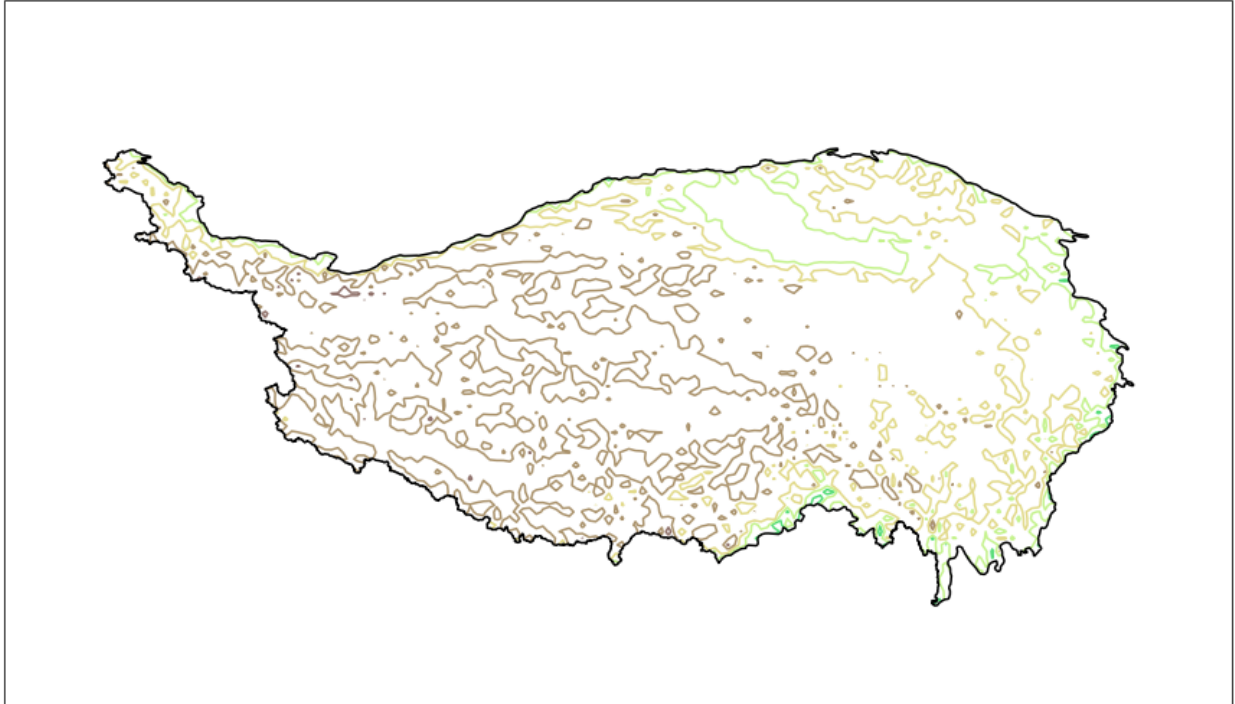
```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_contours_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()
fig = plt.figure(figsize=(18, 9))
fig.tight_layout()

tp = get_map('青藏高原', map_set='geography')

ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
cs = ax.contour(lons, lats, dem, cmap=plt.cm.terrain)
clip_contours_by_map(cs, tp)
draw_map(tp, color='k')
ax.set_extent(tp.get_extent(buffer=3))

plt.show()
```



1.3.8 对 label 的裁减

cnmaps 的 `clip_labels_by_map` 函数可以对超出边界的等值线标签进行裁减。

警告： 由于 Cartopy 自身的设计缺陷，在 0.18.0 版本中，Cartopy 重写的 `clabel` 方法不返回 `Label Text` 对象，因此在该版本中 `clip_labels_by_map` 函数无法生效，在 0.19.0 中修复了这个 bug，所以请尽量使用 0.19.0 及以上版本。

```
import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_contours_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()
fig = plt.figure(figsize=(18, 9))
fig.tight_layout()

tp = get_map('青藏高原', map_set='geography')

ax = fig.add_subplot(111, projection=ccrs.PlateCarree())
cs = ax.contour(lons, lats, dem, cmap=plt.cm.terrain)
clip_contours_by_map(cs, tp)
```

(下页继续)

(续上页)

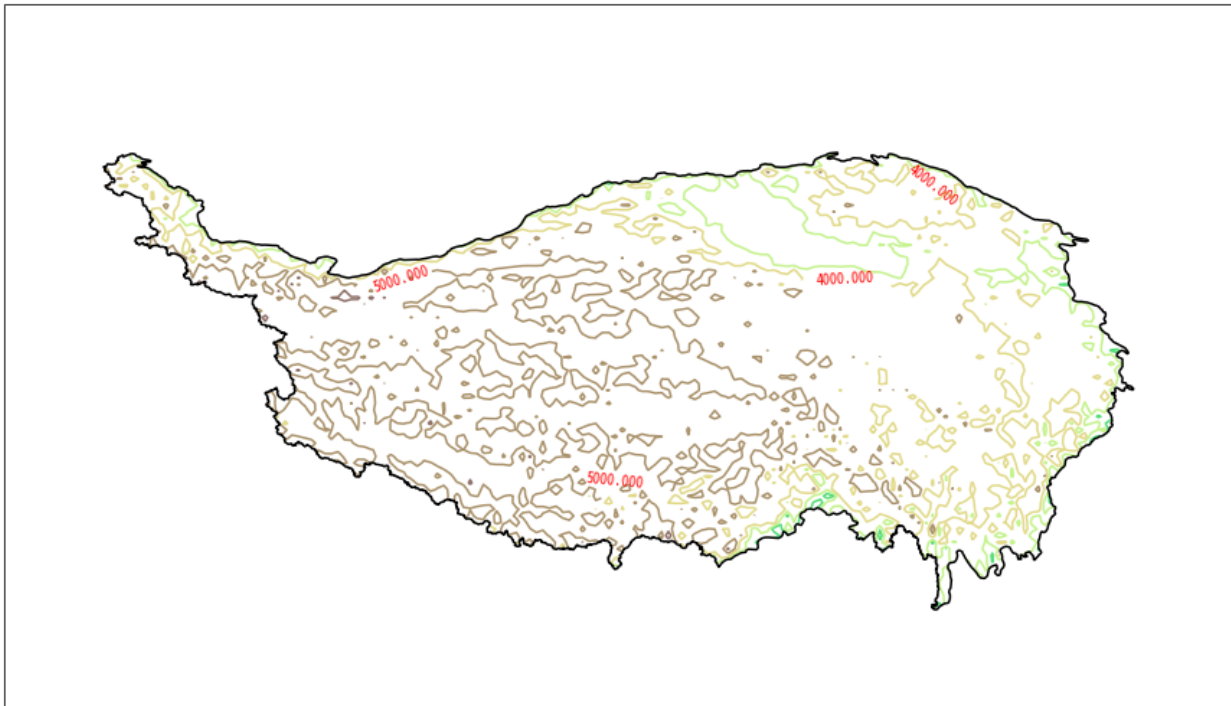
```

cb = ax.clabel(cs, colors='r')
clip_clabels_by_map(cb, tp)

draw_map(tp, color='k')
ax.set_extent(tp.get_extent(buffer=3))

plt.show()

```



1.3.9 变换投影

上述的功能在其他投影下也都适用，我们用四种投影来展示一下变换投影的效果。

```

import cartopy.crs as ccrs
import matplotlib.pyplot as plt
from cnmaps import get_map, draw_map, clip_contours_by_map
from cnmaps.sample import load_dem

lons, lats, dem = load_dem()

PROJECTIONS = [
    ('Mercator', ccrs.Mercator(central_longitude=100)),
    ('Mollweide', ccrs.Mollweide(central_longitude=100)),

```

(下页继续)

```
    ('Orthographic', ccrs.Orthographic(central_longitude=100)),
    ('Robinson', ccrs.Robinson(central_longitude=100))
]

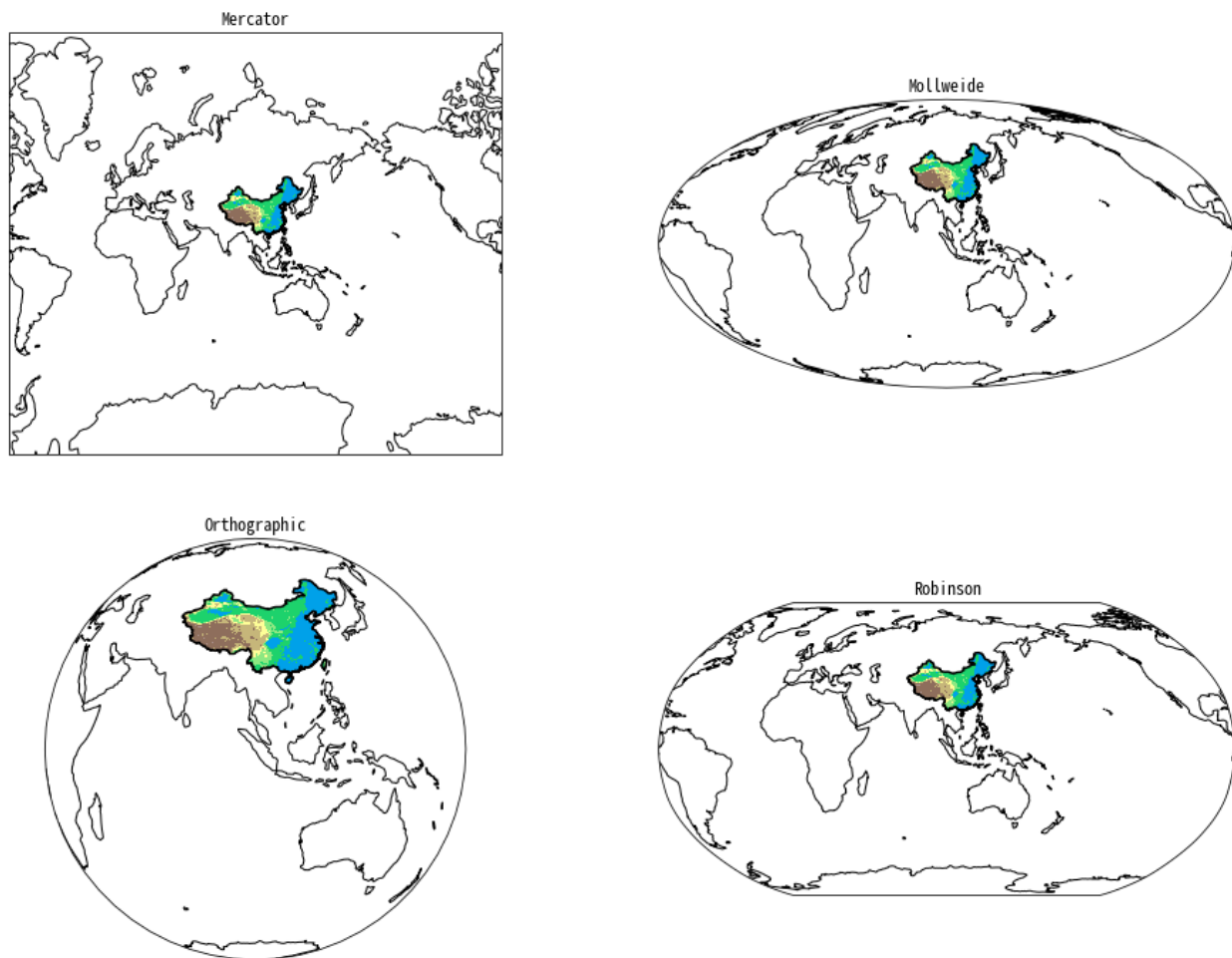
fig = plt.figure(figsize=(16, 12))
fig.tight_layout()

china = get_map('中国')

for i, prj in enumerate(PROJECTIONS):
    ax = fig.add_subplot(2,2,i+1, projection=prj[1])
    cs = ax.contourf(lons, lats, dem, cmap=plt.cm.terrain, transform=ccrs.
↪PlateCarree())
    clip_contours_by_map(cs, china)

    draw_map(china, color='k')
    ax.set_extent(china.get_extent(buffer=3))
    ax.set_global()
    ax.coastlines()
    plt.title(prj[0])

plt.show()
```



1.4 资料引用

1. 谢栋灿. 高德行政区边界获取与整理 (shp 格式) [EB/OL]. [2017.11.05]. <http://i.xdc.at/2017/11/05/amap-district-to-shapefile/>
2. 张懿锂, 李炳元, 郑度. 青藏高原范围与界线地理信息系统数据 [J/DB/OL]. 全球变化数据仓储电子杂志 (中英文), 2014. <https://doi.org/10.3974/geodb.2014.01.12.V1>.

1.5 版本日志

1.5.1 0.2.1

发布时间: 2022-03-02

- 修复了 Windows 系统中 GBK 编码无法加载数据的问题。

1.5.2 0.2.0

发布时间: 2022-02-16

- 增加了对 pcolormesh 图的裁剪支持。
- 修复了边界错误的问题。

1.5.3 0.1.11

发布时间: 2022-02-14

- 尝试修复安装时出现 gbk 编码异常的问题。

1.5.4 0.1.10

发布时间: 2022-02-13

- 增加功能: `cnmaps.get_map` 函数: 获取地图。
- 增加功能: `cnmaps.draw_map` 函数: 绘制地图。
- 增加功能: `cnmaps.MapPolygon` 类: 地图对象, 包括: 加号 (合并)、减号 (剪切)、逻辑与 (交集) 运算符的支持, `get_extent` 方法。
- 增加功能: `cnmaps.clip_contours_by_map` 函数: 基于 `MapPolygon` 类对等值线图做裁减。
- 增加功能: `cnmaps.sample.load_dem` 函数: 加载 dem 样例数据。
- 增加功能: `cnmaps.clip_labels_by_map` 函数: 基于 `MapPolygon` 类对标签做裁减。
- 对 `cartopy.crs` 各类投影的支持。
- 对全国中国国界、全国各省 (特区/直辖市) 地图的预置, 且处理了已知的拓扑错误。
- 集成了 `travis CI` 自动化测试。