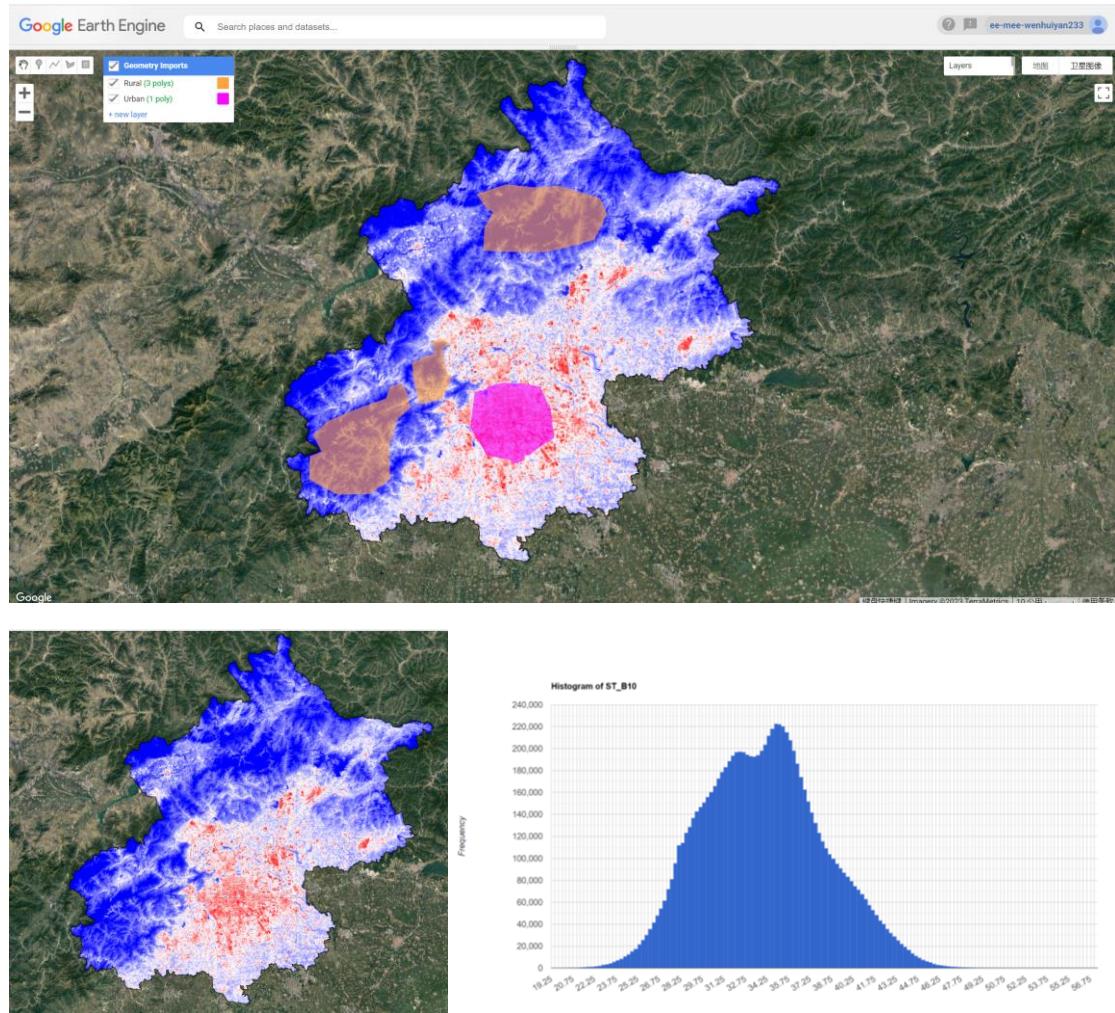


利用 Landsat8 热红外监测北京城市热岛效应

土科 213 闫文慧 2021321010323

- 1>平台: GEE
- 2>数据集: LANDSAT/LC08/C02/T1_L2
- 3>日期: 2013-2023 的北京的七月和八月份
- 4>选择的波段: 'ST_B10', 'QA_PIXEL'
- 5>主要处理: 遮罩云和云阴影; 使用 Landsat 比例因子创建一个函数, 以获取以开尔文和摄氏度为单位的地表温度 (ST)。得到过滤后图像集合并将其显示, 由蓝到白到红, 地表温度越来越高。同时对比了北京城市 (紫色) 和农村 (橙色) 区域的不同, 并生成相关直方图。
- 6>作业链接:
<https://code.earthengine.google.com/30fc40522ee99afa590a7dc47349212>
- 7>成果图展示:



```

Imports (2 entries) ▾
  ↪ var Rural: GeometryCollection ▾
    ↪ var Urban: Polygon, 16 vertices ▾
      ↪ var DATE_RANGE = ee.Filter.date('1883-01-01', '2023-12-31');
      ↪ var YEAR_RANGE = ee.Filter.calendarRange(2013, 2023, 'year');
      ↪ var STUDYBOUNDS = ee.FeatureCollection("projects/eemee-wenhuiyan233/assets/Beijing-2020");
      ↪ Map.setCenterObject(STUDYBOUNDS, 7);
      ↪ Map.setDisplayMode('STUDYBOUNDS');
      ↪ var DISPLAY = true;
      ↪ Map.setOptions('SATELLITE');
      ↪ Map.setCenter([116.46, 39.92], 10); //北京经纬度
      ↪ var LCB_bands = ['ST_B10', 'QA_PIXEL'];

// Landsat 8 和 QA_PIXEL 读取创建一个遮罩云和云阴影的函数//
function cloudMask(image) {
  var qa = image.select('QA_PIXEL');
  var mask = qa.bitwiseAnd(1 << 3);
  var mask = qa.bitwiseAnd(1 << 4); //通过对 qa 进行按位与(btwiseAnd)操作来获取云掩膜，这里的 1 << 3 和 1 << 4 表示对应位数为1的
  return image.updateMask(mask.not()); //将云掩膜应用于输入的影像数据，返回掩膜后的影像数据。
}

var L8 = ee.ImageCollection('LANDSAT/LC08/C02/T1_L2')
  .select('ST_B10', 'QA_PIXEL')
  .filterBounds(STUDYBOUNDS)
  .filter(DATE_RANGE)
  .filter(YEAR_RANGE)
  .map(cloudMask); //应用云掩膜函数 cloudMask 对该图像集合中的每张影像进行处理。
var filtered_L8 = L8.filter(ee.Filter.lt('CLOUD_COVER', 20)); //筛选云覆盖率大于等于 20% 的影像数据。
print(filtered_L8, "Landsat 8 ST"); //打印筛选后的 Landsat 8 影像数据集合 filtered_L8，名为 Landsat 8 ST.

/*, 用于导出开尔文和摄氏度的温度*/
function applyScaleFactors(image) {
  var thermalBands = image.select('ST_B10').multiply(0.00341802).add(149.0) // 选择 ST_B10 波段，将其乘以比例因子 0.00341802。
  .subtract(273.15); //从上面计算出的 Kelvin 温度值中减去 273.15，将其转换为以摄氏度为单位的温度值。
  return image.addBands(thermalBands, null, true); //将新的温度波段添加到图像数据中，并将同名现有波段替换为新的波段。
}
print(filtered_L8, "Landsat 8 (Celsius)"); //输出经过处理后的 Landsat 影像数据，其中包括地表温度数据，单位是摄氏度。
var landsatST = filtered_L8.map(applyScaleFactors); //定义一个变量，将比例因子应用于过滤后的影像集合。
print(filtered_L8, "... Computing mean ST across image collection"); //在控制台选项卡中跟踪进度。

//计算平均地表温度 *****
//整个过滤后的图像集合。
var mean_LandsatST = landsatST.mean(); //定义一个变量来计算每个像素的ST平均表面温度
var clip_mean_ST = mean_LandsatST.clip(STUDYBOUNDS);
print(clip_mean_ST, 'Mean ST clipped to study area');
var values_ST = clip_mean_ST.select('ST_B10'); // 只选择表面温度值
var histogram_ST_values = ui.Chart.image.histogram(values_ST, STUDYBOUNDS, 48); //输出aoi中平均ST值的直方图。
print(histogram_ST_values); //使用print语句输出平均ST的直方图值到控制台选项卡

50
51 //添加图像到地图窗口，定义最小/最大值
52 Map.addLayer(clip_mean_ST, {
53   bands: "ST_B10",
54   min: 28, max: 47,
55   palette: ["blue", "white", "red"], "ST", DISPLAY);
56
57
58 print('... Mean/Min/Max ST for urban Beijing');//为可视化指定一个名称，并显示结果。
59 var statsUrban = clip_mean_ST.reduceRegions({
60   collection: Urban,
61   reducer: ee.Reducer.mean().combine({//两个核计算共享相同的输入数据。这样可以避免重复计算，提高代码运行的效率。
62     type: 'mean', scale: 30}), //空间分辨率
63   reducer2: ee.Reducer.minMax(),
64   sharedInputs: true
65 },
66   scale: 30}); //空间分辨率
67 print(statsUrban);
68
69 print('... Mean/Min/Max ST for rural Beijing');
70 var statsRural = clip_mean_ST.reduceRegions({
71   collection: Rural,
72   reducer: ee.Reducer.mean().combine({
73     reducer2: ee.Reducer.minMax(),
74     sharedInputs: true
75   }),
76   scale: 30});
77 print(statsRural);
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101

```

Use print(...) to write to this console.

- ImageCollection LANDSAT/LC08/C02/T1_L2 (91 elements)
JSON
Landsat 8 ST
JSON
- ImageCollection LANDSAT/LC08/C02/T1_L2 (91 elements)
JSON
Landsat ST (Celsius)
JSON
- ... Computing mean ST across image collection
JSON
- Image (2 bands)
Mean ST clipped to study area
JSON
JSON

Histogram of ST_B10

Frequency

18.25 23.25 28.25 33.25 38.25 43.25 48.25 53.25 58.25

... Mean/Min/Max ST for urban Beijing
JSON

FeatureCollection (1 element, 7 columns)
JSON

... Mean/Min/Max ST for rural Beijing
JSON

FeatureCollection (1 element, 7 columns)
JSON

Use print(...) to write to this console.

- ImageCollection LANDSAT/LC08/C02/T1_L2 (91 elements)
JSON
Landsat 8 ST
JSON
- ImageCollection LANDSAT/LC08/C02/T1_L2 (91 elements)
JSON
Landsat ST (Celsius)
JSON
- ... Computing mean ST across image collection
JSON
- Image (2 bands)
Mean ST clipped to study area
JSON
JSON

Histogram of ST_B10

Frequency

18.25 23.25 28.25 33.25 38.25 43.25 48.25 53.25 58.25

... Mean/Min/Max ST for urban Beijing
JSON

FeatureCollection (1 element, 7 columns)
JSON

... Mean/Min/Max ST for rural Beijing
JSON

FeatureCollection (1 element, 7 columns)
JSON