05_Flood_risk_model

October 14, 2020

1 Flood risk modelling based on DTM

Computational notebook 05 for Climate adaptation plans in the context of coastal settlements: the case of Portugal.

Date: 27/06/2020

This notebooks models floor risk under "what if" +5 m scenario based on digital terrain model on Portuguese coast provided by Direção-Geral do Território (DGT) - `Modelo Digital do Terreno das Zonas Costeiras de Portugal Continental com resolução de 1 m (400 m em terra) - LiDAR, 2011-12-07`. Unfortunately, we do not hold rights to share the data within this repository.

DTM data are stored in separate ASC files based on the grid defined in `MDT1m_LiDAR2011_secciona.shp`.

`name_tess` layers require additional attribute `main` to be set to 1 for cells in the first row. Note that this information was not used in the final manuscript.

Structure of data:

```
./data/
    atlantic.gpkg
        name_blg    - Polygon layers
        name_str    - LineString layers
        name_case   - Polygon layers
        name_tess   - Polygon layers
        name_blocks - Polygon layers
        ...
     preatl.gpkg
        name_blg
        name_str
        name_case
        ...
     premed.gpkg
        name_blg
        name_str
        name_case
        ...
```

```
    med.gpkg
       name_blg
       name_str
       name_case
       ...
./MDT/
    MDT1m_LiDAR2011_secciona.shp
    002_1_55-top_orto.asc
    002_2_53-top_orto.asc
    ...
```

CRS of the original data is EPSG:3763.

```
<Projected CRS: EPSG:3763>
Name: ETRS89 / Portugal TM06
Axis Info [cartesian]:
- X[east]: Easting (metre)
- Y[north]: Northing (metre)
Area of Use:
- name: Portugal - mainland - onshore
- bounds: (-9.56, 36.95, -6.19, 42.16)
Coordinate Operation:
- name: Portugal TM06
- method: Transverse Mercator
Datum: European Terrestrial Reference System 1989
- Ellipsoid: GRS 1980
- Prime Meridian: Greenwich
```

[1]:
```
import geopandas as gpd
import rasterio as rio
from rasterio.merge import merge
import rasterstats
import pandas as pd
import numpy as np
import fiona
```

[2]:
```
fiona.__version__, gpd.__version__, rio.__version__, rasterstats.__version__,␣
↪np.__version__, pd.__version__
```

[2]:
```
('1.8.13', '0.7.0', '1.1.3', '0.14.0', '1.18.1', '1.0.3')
```

[174]:
```
grid = gpd.read_file('MDT/MDT1m_LiDAR2011_secciona.shp')
```

[197]:
```
folder = '/Users/martin/Dropbox/Academia/Data/Geo/Portugal/'
```

[201]:
```
parts = ['atlantic', 'preatl', 'premed', 'med']
for part in parts:
    path = folder + part + '.gpkg'
    layers = [x[:-4] for x in fiona.listlayers(path) if 'blg' in x]
```

```
    for l in layers:
        blg = gpd.read_file(path, layer=l + '_blg')
        case = gpd.read_file(path, layer=l + '_case')

        rparts = grid[grid.intersects(case.unary_union)].Id_Unidade
        rasters = []
        for rpart in rparts:
            rpath = 'MDT/' + rpart + '-top_orto.asc'
            rasters.append(rio.open(rpath))
        if len(rasters) > 1:
            array, affine = merge(rasters)
            stats = rasterstats.zonal_stats(blg, array[0], affine=affine,␣
↪stats=['min', 'max', 'median', 'mean', 'count'])
        else:
            stats = rasterstats.zonal_stats(blg, rasters[0].read(1),␣
↪affine=rasters[0].transform, stats=['min', 'max', 'median', 'mean', 'count'])
        if 'min' in blg.columns:
            blg = blg.drop(columns=['min', 'max', 'median', 'mean', 'count'])
        blg = blg.join(pd.DataFrame(stats))
        blg.to_file(path, layer=l + '_blg', driver='GPKG')
        print(part, l, 'done')
```

```
atlantic foz done
atlantic aguda done
atlantic esposende done
atlantic espinho done
atlantic povoa done
atlantic vila_do_conde done
atlantic vila_praia done
preatl costa_nova done
preatl praia_mira done
preatl palheiros done
preatl figueira_foz done
preatl pedrogao done
preatl sao_martinho done
preatl ericeira done
preatl azenhas done
preatl quaios done
preatl vieira done
preatl foradouro done
preatl nazare done
premed cascais done
premed costa_caparica done
premed praia_corvoeiro done
premed zambujeira done
premed sesimbra done
med armacao done
```

```
med olhao done
med monte_gordo done
med albufeira done
med faro done
med quarteira done
```

[210]:
```
parts = ['atlantic', 'preatl', 'premed', 'med']
for part in parts:
    path = folder + part + '.gpkg'
    layers = [x[:-4] for x in fiona.listlayers(path) if 'blg' in x]
    for l in layers:
        blg = gpd.read_file(path, layer=l + '_blg')
        blg = blg.replace(-999, np.nan)
        print(part, l, '- NaN in min:', blg['min'].isna().sum(), '/', len(blg))
```

```
atlantic foz - NaN in min: 0 / 482
atlantic aguda - NaN in min: 0 / 255
atlantic esposende - NaN in min: 0 / 81
atlantic espinho - NaN in min: 0 / 485
atlantic povoa - NaN in min: 0 / 393
atlantic vila_do_conde - NaN in min: 0 / 320
atlantic vila_praia - NaN in min: 0 / 197
preatl costa_nova - NaN in min: 0 / 560
preatl praia_mira - NaN in min: 0 / 496
preatl palheiros - NaN in min: 0 / 176
preatl figueira_foz - NaN in min: 232 / 842
preatl pedrogao - NaN in min: 0 / 233
preatl sao_martinho - NaN in min: 0 / 150
preatl ericeira - NaN in min: 0 / 258
preatl azenhas - NaN in min: 0 / 65
preatl quaios - NaN in min: 0 / 200
preatl vieira - NaN in min: 0 / 469
preatl foradouro - NaN in min: 0 / 238
preatl nazare - NaN in min: 0 / 738
premed cascais - NaN in min: 0 / 273
premed costa_caparica - NaN in min: 0 / 488
premed praia_corvoeiro - NaN in min: 0 / 277
premed zambujeira - NaN in min: 0 / 286
premed sesimbra - NaN in min: 0 / 426
med armacao - NaN in min: 0 / 401
med olhao - NaN in min: 146 / 146
med monte_gordo - NaN in min: 2 / 103
med albufeira - NaN in min: 0 / 383
med faro - NaN in min: 279 / 279
med quarteira - NaN in min: 0 / 269
```

[213]:
```
parts = ['atlantic', 'preatl', 'premed', 'med']
for part in parts:
    path = folder + part + '.gpkg'
    layers = [x[:-4] for x in fiona.listlayers(path) if 'blg' in x]
    for l in layers:
        blg = gpd.read_file(path, layer=l + '_blg')
        blg = blg.replace(-999, np.nan)
        print(part, l, (blg['min']< 5).sum(), (blg['min']< 5).sum() / len(blg))
```

```
atlantic foz 0 0.0
atlantic aguda 2 0.00784313725490196
atlantic esposende 20 0.24691358024691357
atlantic espinho 0 0.0
atlantic povoa 27 0.06870229007633588
atlantic vila_do_conde 0 0.0
atlantic vila_praia 7 0.03553299492385787
preatl costa_nova 324 0.5785714285714286
preatl praia_mira 40 0.08064516129032258
preatl palheiros 2 0.011363636363636364
preatl figueira_foz 67 0.07957244655581948
preatl pedrogao 0 0.0
preatl sao_martinho 136 0.9066666666666666
preatl ericeira 0 0.0
preatl azenhas 0 0.0
preatl quaios 0 0.0
preatl vieira 136 0.2899786780383795
preatl foradouro 79 0.3319327731092437
preatl nazare 74 0.1002710027100271
premed cascais 69 0.25274725274725274
premed costa_caparica 362 0.7418032786885246
premed praia_corvoeiro 4 0.01444043321299639
premed zambujeira 0 0.0
premed sesimbra 33 0.07746478873239436
med armacao 112 0.2793017456359102
med olhao 0 0.0
med monte_gordo 52 0.5048543689320388
med albufeira 9 0.02349869451697128
med faro 0 0.0
med quarteira 110 0.40892193308550184
```

[249]:
```
waterrelation = pd.DataFrame()
```

[251]:
```
parts = ['atlantic', 'preatl', 'premed', 'med']
for part in parts:
    path = folder + part + '.gpkg'
    layers = [x[:-4] for x in fiona.listlayers(path) if 'blg' in x]
    for l in layers:
```

```
        buildings = gpd.read_file(path, layer=l + '_blg')
        tessellation = gpd.read_file(path, layer=l + '_tess')

        buildings = buildings.merge(tessellation[['uID', 'main']], on='uID',␣
↪how='left')
        if 'main' not in buildings.columns:
            import warnings
            warnings.warn(l)
        main = buildings[buildings['main'] == 1]

        if 'part' in main.columns:
            for part in set(main.part):
                subset = main.loc[main.part == part]
                mainset = buildings.loc[buildings.part == part]

                waterrelation.loc[l + str(part), 'min' + '_min'] =␣
↪subset['min'].min()
                waterrelation.loc[l + str(part), 'min' + '_med'] =␣
↪subset['min'].median()
                waterrelation.loc[l + str(part), 'flooded_perc'] =␣
↪(mainset['min']< 5).sum() / len(mainset)

        else:
            waterrelation.loc[l, 'min' + '_min'] = main['min'].min()
            waterrelation.loc[l, 'min' + '_med'] = main['min'].median()
            waterrelation.loc[l, 'flooded_perc'] = (buildings['min']< 5).sum() /
↪ len(buildings)
```

[253]:
```
waterrelation.to_csv('data/waterrelation_data.csv')
```