

mountain_avalanche

August 7, 2020

```
[1]: # importing standard python libraries + geopandas for dealing with geospatial_
      ↪data
import geopandas as gpd
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
[2]: # reading shapefile of France divided into communes
gdf = gpd.
      ↪read_file(r"CONTOURS-IRIS\1_DONNEES_LIVRAISON_2014\CONTOURS-IRIS_2-0_SHP_LAMB93_FE-2014\CON
      ↪shp")
```

```
[3]: # viewing geodataframe
gdf.head()
```

```
[3]:  DEPCOM      NOM_COM  IRIS  DCOMIRIS      NOM_IRIS  TYP_IRIS  \
0  95580      Saint-Witz  0000  955800000      Saint-Witz      Z
1  95258      Frouville  0000  952580000      Frouville      Z
2  95116  Bruyères-sur-Oise  0000  951160000  Bruyères-sur-Oise      Z
3  95308      Hérouville  0000  953080000      Hérouville      Z
4  95055  Bellefontaine  0000  950550000  Bellefontaine      Z
```

```
                                geometry
0  POLYGON ((667826.500 6888732.700, 667871.500 6...
1  POLYGON ((639020.100 6896582.300, 638964.900 6...
2  POLYGON ((650549.700 6898511.300, 650534.500 6...
3  POLYGON ((635467.800 6888554.700, 635389.470 6...
4  POLYGON ((660098.000 6887893.200, 660133.100 6...
```

```
[4]: # reading Excel sheet with number of avalanche accidents in communes
avalanche = pd.read_excel("tableau-accidents.xlsx")
avalanche.head()
```

```
[4]:  code accident      date  heure  département      commune  massif  \
0    1011-05-01  2011-01-08  11:06              5  monetier les bains  pelvoux
1    1011-05-02  2011-01-30  10:00              5             molines  queyras
```

2	1011-05-03	2011-02-19	13:10	5	orcières	champsaur
3	1011-05-04	2011-02-26	16:10	5	orcières	champsaur
4	1011-05-05	2011-03-02	16:40	5	val des prés	queyras

	site	coordonnées	nzone	départ	altitude
0	cibouit			NaN	NaN
1	col agnel	32t0339842	4950312		2650.0
2	col de la montagne haute / mourre froid			NaN	2800.0
3	col combeau / grande autane	32t 0292853	4947510		2100.0
4	vallon de Vachette / secteur ombilic			NaN	2005.0

	inclinaison	... évolution	BRA	qualité	nneige	cohésion	nneige
0	NaN	...	descente	3	sèche		tendre
1	35-39	...	montée	2	sèche		tendre
2	35-39	...	montée	3	sèche		tendre
3	40-44	...	descente	3	sèche		tendre
4	35-39	...	descente	3	sèche		tendre

	type	ndépart	cause	départ	dénivelé	n(mètres)
0	linéaire	accidentelle	soi-même		NaN	
1	linéaire	accidentelle	soi-même		30	
2	linéaire	accidentelle	soi-même		200	
3	linéaire		naturelle		100	
4	linéaire	accidentelle	soi-même		500	

	largeur	ncassure	n(mètres)	épaisseur	ncassure max.	n(cm)	Unnamed: 27
0			100.0			50.0	NaN
1			40.0			100.0	NaN
2			150.0			80.0	NaN
3			5.0			30.0	NaN
4			60.0			60.0	NaN

[5 rows x 28 columns]

```
[5]: # names of communes in geodataframe gdf and dataframe avalanche need to be in
      ↪ same format
      # avoiding mixing upper and lowercase
      avalanche['commune'] = avalanche.commune.str.lower()
      gdf['NOM_COM'] = gdf.NOM_COM.str.lower()
```

```
[6]: # eyeballing through manually inserted names of communes I found some double
      ↪ records
      # for ex: 'bagnères de luchon', 'bareges', 'barèges'
      # or "val d'isere", "val d'isère", 'val isere', 'val isère'
      np.sort(avalanche.commune.unique())
```

```
[6]: array(['abondance', 'abriès', 'aillons le vieux', 'aime', 'allemond',
'allos', 'alpe d'huez', 'araches la frasse', 'aragnouet',
'arvieux', 'arêche beaufor', 'arêche beaufort', 'aston', 'aulon',
'auris', 'auris en oisans', 'aussois', 'auzat', 'avoriaz',
'avrieux', 'ax les thermes', 'bagnère de bigorre',
'bagnères de bigorre', 'bagnères de luchon', 'bareges', 'barèges',
'beaufort', 'bellentre', 'bellevaux', 'bernex', 'bessans',
'bezaudun sur bine', 'bonneval', 'bonneval s/ arc',
'bonneval sur arc', 'bourg saint maurice', 'bourg st maurice',
'boutx', 'cauteret', 'cauterets', 'ceillac', 'cervièrès',
'chamonix', 'champagny', 'champagny en vanoise', 'champcella',
'chamrousse', 'chantelouve', 'chateauroux les alpes', 'chatel',
'chevaline', 'contamines', 'contamines montjoie', 'cordon',
'corrençon en vercors', 'cote d'aime', 'courchevel', 'crots',
'crévoux', 'doucy en bauges', 'dévoluy', 'eaux bonnes',
'enchastayes', 'entraunes', 'err', 'favergeres-seythenex',
'ferrère', 'fontcouverte la toussuire', 'formigueres',
'freissinière', 'freney d'oisans', 'gavarnie', 'gedre', 'glandage',
'gouaux de larboust', 'granier', 'guillestre', 'hauteluze',
'isola', 'jarsy', 'jausiers', 'l'hospitalet pres l'andorre',
'la bathie', 'la chapelle en valgaudemar', 'la clusaz',
'la combe de lancey', 'la cote d'aime', 'la ferrière',
'la ferrière d'allevard', 'la giettaz', 'la grave', 'la lèche',
'la morte', 'la perrière', 'la plagne tarentaise',
'la salle les alpes', 'la vigerie', 'landry', 'lanslebourg',
'lanslevillard', 'laruns', 'laval', 'laval dens', 'laveissière',
'le bouchet montcharvin', 'le chatel', 'le freney d'oisans',
'le freneyr d'oisans', 'le grand bornand', 'le monetier les bains',
'le mont dore', 'le monétier les bains',
'le petit bornand des glières', 'le reposoir', 'le vernet',
'les adrets', 'les allues', 'les avanchers',
'les avanchers-valmorel', 'les belleville', 'les clefs',
'les contaminés', 'les contaminés montjoie', 'les deux alpes',
'les houches', 'les orres', 'lozzi', 'macot', 'macot la plagne',
'magland', 'manigod', 'metzeral', 'mieussy', 'mittlach', 'modane',
'molines', 'molines en queyras', 'monetier', 'monetier les bains',
'mont de lans', 'mont dore', 'montchavin', 'montclar',
'montgellafrey', 'montgenevre', 'montgenèvre', 'montricher albane',
'montricher albanne', 'montricher-albanne', 'montsapey',
'montvalezan', 'monétier les bains', 'morzine', 'méribel',
'névache', 'ombleze', 'orcières', 'orcières', 'orelle', 'ornon',
'oz en oisans', 'oô', 'peisey', 'peisey nancroix',
'peisey-nancroix', 'pelvoux', 'pinsot', 'pontamafrey montpascal',
'porta', 'porté puymorens', 'pralognan', 'pralognan la vanoise',
'praz sur arly', 'presle', 'puy saint vincent', 'puy st vincent',
'rabat les trois seigneurs', 'revel', 'risoul', 'ristolas',
'réalion', 'saint agnès', 'saint andré d'embrun',
```

```

'saint bon courchevel', 'saint bon tarentaise', 'saint chaffrey',
'saint christophe en oisans', 'saint christophe sur guiers',
'saint colomban les villards', 'saint crepin',
'saint dalmas de selvage', 'saint etienne de tinee',
'saint gervais', 'saint gervais les bains', 'saint honoré',
'saint jean de belleville', 'saint martin belleville',
'saint martin de belleville',
'saint martin de belleville (val thorens)',
'saint maurice sur moselle', 'saint pancrasse',
'saint paul s/ ubaye', 'saint paul sur ubaye',
'saint pierre de chartreuse', 'saint veran', 'saint véran',
'sainte agnes', 'sainte foy tarentaise', 'sainte marie de cuines',
'samoens', 'sarcenas', 'serraval', 'sers', 'seythenex', 'sixt',
'sixt fer à cheval', 'st bernard du touvet', 'st bon tarentaise',
'st christophe en oisans', 'st christophe sur guiers',
'st dalmas le selvage', 'st etienne dévoluy', 'st etienne tinee',
'st françois longchamp', 'st gervais les bains', 'st honoré',
'st lary soulan', 'st martin de belleville', 'st martin vésubie',
'st pierre de chartreuse', 'st véran', 'ste foy tarentaise',
'stockersohn', 'stosswir', 'stosswihr', 'termignon',
'thollon les mémises', 'tignes', 'ustou', 'uvernnet fours',
'vacheresse', "val d'isere", "val d'isère", 'val des prés',
'val isere', 'val isère', 'valloire', 'vallorcine', 'valmeinier',
'vars', 'vaujany', 'vieille aure', 'villar arene',
'villard d'arene', 'villard de lans', 'villards de lans',
'villarodin bourget', 'villaroger', 'wasserbourg', 'épierre'],
dtype=object)

```

```

[7]: # because of inconsistent using of french special characters and some other
      ↪ characters I change them to unified writing
avalanche['commune'].replace(['è', 'é', 'ê', 'ë'], 'e', inplace=True,
      ↪ regex=True)
avalanche['commune'].replace('-', ' ', inplace=True, regex=True)
avalanche['commune'].replace('à', 'a', inplace=True, regex=True)
avalanche['commune'].replace('î', 'i', inplace=True, regex=True)
avalanche['commune'].replace('ô', 'o', inplace=True, regex=True)
avalanche['commune'].replace("d'", "d ", inplace=True, regex=True)
avalanche['commune'].replace(['st ', 'St '], 'saint ', inplace=True, regex=True)
avalanche['commune'].replace('s/', 'sur', inplace=True, regex=True)
gdf['NOM_COM'].replace(['è', 'é', 'ê', 'ë'], 'e', inplace=True, regex=True)
gdf['NOM_COM'].replace('-', ' ', inplace=True, regex=True)
gdf['NOM_COM'].replace(['à', 'â'], 'a', inplace=True, regex=True)
gdf['NOM_COM'].replace('î', 'i', inplace=True, regex=True)
gdf['NOM_COM'].replace('ô', 'o', inplace=True, regex=True)
gdf['NOM_COM'].replace("d'", "d ", inplace=True, regex=True)
gdf['NOM_COM'].replace(['st ', 'St '], 'saint ', inplace=True, regex=True)
gdf['NOM_COM'].replace('s/', 'sur', inplace=True, regex=True)

```

```
[8]: # all commune names in avalanche communes should be also in gdf geodataframe
# to find avalanche communes missing from gdf because of use of inconsistent
# names
# I will merge both dataframes and check differences
aval_gdf = pd.merge(gdf, avalanche, left_on='NOM_COM', right_on='commune',
# how='inner')
# creating two arrays of commune names in merged dataframe and avalanche
# dataframe
aval_gdf_com = np.sort(aval_gdf['NOM_COM'].unique())
avalanche_com = np.sort(avalanche['commune'].unique())
# looking for communes in avalanche dataframe not found in merged aval_gdf
np.setdiff1d(avalanche_com, aval_gdf_com)
```

```
[8]: array(['aillons le vieux', 'alpe d huez', 'areche beaufor',
'areche beaufort', 'auris en oisans', 'avoriaz',
'bagnere de bigorre', 'cauteret', 'chamonix', 'contamines',
'contamines montjoie', 'cote d aime', 'courchevel',
'faverges seythenex', 'freissiniere', 'freney d oisans',
'la ferriere d allevard', 'la plagne tarentaise', 'la vigerie',
'lanslebourg', 'le bouchet montcharvin', 'le freneyr d oisans',
'le mont dore', 'le petit bornand des glieres', 'les avanchers',
'les belleville', 'les contaminés', 'les deux alpes', 'macot',
'meribel', 'moline', 'monetier', 'monetier les bains',
'montchavin', 'montricher albane', 'oz en oisans', 'peisey',
'pralognan', 'saint agnes', 'saint bernard du touvet',
'saint bon courchevel', 'saint colomban les villards',
'saint dalmas de selvage', 'saint etienne devoluy',
'saint etienne tinee', 'saint martin belleville',
'saint martin de belleville (val thorens)', 'sixt',
'ste foy tarentaise', 'stockersohn', 'stosswir', 'val isere',
'vieille aure', 'villar arene', 'villard d arene',
'villards de lans'], dtype=object)
```

```
[9]: # I take 'alpe d huez' as example. Why it was not found in aval_gdf_com?
# correct name for commune is not 'alpe d huez', but just 'huez'
gdf[gdf.NOM_COM.str.contains(pat='crepin')]
```

```
[9]:
```

	DEPCOM	NOM_COM	IRIS	DCOMIRIS \
18745	61339	putanges pont ecrepin	0000	613390000
19604	60570	saint crepin ibouvillers	0000	605700000
19857	60569	saint crepin aux bois	0000	605690000
39330	24391	saint crepin de richemont	0000	243910000
39378	24392	saint crepin et carlucet	0000	243920000
39742	24390	saint crepin d auberoche	0000	243900000
42421	17321	saint crepin	0000	173210000
48084	05136	saint crepin	0000	051360000

	NOM_IRIS	TYP_IRIS	\
18745	Putanges-Pont-Écrepin	Z	
19604	Saint-Crépin-Ibouwillers	Z	
19857	Saint-Crépin-aux-Bois	Z	
39330	Saint-Crépin-de-Richemont	Z	
39378	Saint-Crépin-et-Carlucet	Z	
39742	Saint-Crépin-d'Auberoche	Z	
42421	Saint-Crépin	Z	
48084	Saint-Crépin	Z	

	geometry
18745	POLYGON ((459538.600 6857933.400, 459568.600 6...
19604	POLYGON ((632796.600 6909698.000, 632890.520 6...
19857	POLYGON ((701678.400 6927283.400, 701640.970 6...
39330	POLYGON ((513735.100 6486445.400, 513777.130 6...
39378	POLYGON ((565960.500 6427850.000, 565881.880 6...
39742	POLYGON ((533502.200 6450714.900, 533502.500 6...
42421	POLYGON ((407928.800 6550148.800, 407962.220 6...
48084	POLYGON ((982424.700 6406169.100, 982434.000 6...

```
[10]: # I will create dictionary with wrong commune names as keys and its correct
      ↪ variants as values
# based on similar search as in previous example and wikipedia info about
      ↪ changes of commune names
correct = {'aillons le vieux':'aillon le vieux', 'alpe d huez':'huez', 'areche
      ↪ beaufort':'beaufort savoie',
          'areche beaufort':'beaufort savoie','auris en oisans':'auris', 'avoriaz':
      ↪ 'morzine',
          'bagnere de bigorre':'bagneres de bigorre', 'cauteret':'cauterets',
      ↪ 'chamonix':'chamonix mont blanc',
          'contamines':'les contaminés montjoie', 'contamines montjoie':'les
      ↪ contaminés montjoie',
          'cote d aime':'la plagne tarentaise', 'freissiniere':'freissinieres',
      ↪ 'freney d oisans':'le freney d oisans',
          'la ferriere d allevard':'haut breda', 'la ferriere':'haut breda', 'la
      ↪ vigerie':'lavigerie', 'la perriere':'courchevel',
          'lanslebourg':'lanslebourg mont cenis', 'le bouchet montcharvin':'le
      ↪ bouchet',
          'le freneyr d oisans':'le freney d oisans', 'le mont dore':'mont dore',
          'le petit bornand des glières':'le petit bornand les glières',
          'les avanchers':'les avanchers valmorel','les contaminés':'les
      ↪ contaminés montjoie',
          'macot':'la plagne tarentaise', 'meribel':'les allues', 'moline':
      ↪ 'moline en queyras',
          'monetier':'le monetier les bains', 'monetier les bains':'le monetier
      ↪ les bains',
```

```

'montchavin':'la plagne tarentaise', 'montricher albane':'montricher_
↳albanne', 'oz en oisans':'oz',
'peisey':'peisey nancroix', 'pralognan':'pralognan la vanoise', 'saint_
↳agnes':'sainte agnes isere',
'saint bernard du touvet':'le touvet', 'saint bon courchevel':
↳'courchevel',
'saint colomban les villards':'saint colomban des villards', 'saint_
↳dalmas de selvage':'saint dalmas le selvage',
'saint etienne devoluy':'devoluy', 'saint etienne tinee':'saint etienne_
↳de tinee', 'saint gervais':'saint gervais les bains',
'saint martin belleville':'les belleville', 'saint martin de belleville_
↳(val thorens)':'les belleville',
'saint martin de belleville':'les belleville', 'saint crepin':'saint_
↳crepin alpes',
'sixt':'sixt fer a cheval', 'ste foy tarentaise':'sainte foy_
↳tarentaise', 'sers':'sers haute bigorre',
'stockersohn':'storckensohn', 'stosswhir':'stosswhir',
'val isere':'val d isere', 'vieille aure':'', 'villar arene':'villar d_
↳arene', 'villard d arene':'villard d arene',
'villards de lans':'villard de lans', 'beaufort':'beaufort savoie',_
↳'bonneval':'bonneval sur arc', 'vars':'vars hautes alpes'}

```

```

[11]: # loop to change wrong commune names, replacing dictionary key by value
correct_commune = []
for commune in avalanche.commune:
    if commune in list(correct.keys()):
        correct_commune.append(correct[commune])
    else:
        correct_commune.append(commune)

```

```

[12]: # checking column with corrected names is same length as original
(len(avalanche['commune'])) == (len(correct_commune))

```

[12]: True

```

[13]: # adding column with correct column names
avalanche['commune'] = correct_commune

```

```

[14]: # some of commune names in gdf are outdated or same name is used for more_
↳communes
# therefore I create dictionary to correct them based on DEPCOM value

gdf_correct = {'73034':'beaufort savoie', '38350':'sainte agnes isere',
'38163':'haut breda', '05177':'vars hautes alpes', '65046':_
↳'aulon',
'65424': 'sers haute bigorre', '05027':'cervieres hautes alpes',

```

```

        '73227': 'courchevel', '73150': 'la plagne tarentaise', '38253':
        ↪ 'les deux alpes',
        '73257': 'les belleville', '76589': 'saint honore normandie',
        '04237': 'le vernet', '39747': 'sainte agnes jura',
        '05136': 'saint crepin alpes', '06113': 'sainte agnes pres nice',
        '31451': 'revel occitanie', '53130': 'laval mayenne',
        '11242': 'montclar occitanie', '12149': 'montclar occitanie 2',
        '74123': 'faverge seythenex', '23011': 'aulon aquitaine',
        '31023': 'aulon occitanie', '73198': 'courchevel',
        '16393': 'vars aquitaine', '70523': 'vars franche comte',
        '31051': 'beaufort occitanie', '34026': 'beaufort occitanie_
        ↪ beziers',
        '38032': 'beaufort isere', '39043': 'beaufort orbagna', '59058':
        ↪ 'beaufort nord',
        '30256': 'saint gervais occitanie', '33415': 'saint gervais_
        ↪ aquitanie',
        '38390': 'saint gervais isere', '95554': 'saint gervais ile de_
        ↪ france',
        '22058': 'plemet', '37106': 'la ferriere loire', '85089': 'la_
        ↪ ferrierre nantes',
        '28051': 'bonneval loire', '43035': 'bonneval haute loire',_
        ↪ '73046': 'la lechere',
        '03306': 'le vernet pres vichy', '09331': 'le vernet occitanie',_
        ↪ '43260': 'le vernet haute loire',
        '73198': 'la perriere old', '73046': 'bonneval old', '05177': 'vars_
        ↪ hautes alpes'}

# for loop to correct commune names
correct_commune_gdf = []
i = -1
for depcom in gdf.DEPCOM:
    i += 1
    if depcom in list(gdf_correct.keys()):
        correct_commune_gdf.append(gdf_correct[depcom])
    else:
        correct_commune_gdf.append(gdf.NOM_COM[i])

```

```

[15]: # checking column with corrected names is same length as original
      (len(gdf['NOM_COM'])) == (len(correct_commune_gdf))

```

[15]: True

```

[16]: # adding column with correct commune names
      gdf['NOM_COM'] = correct_commune_gdf

```



```
[17]: aval_gdf = pd.merge(gdf, avalanche, left_on='NOM_COM', right_on='commune',
    ↪how='inner')
aval_gdf_com = np.sort(aval_gdf['NOM_COM'].unique())
avalanche_com = np.sort(avalanche['commune'].unique())
```

```
[18]: # check if there are still some communes missing from merged aval_gdf
# actually there are all old names of communes which need to be corrected to
    ↪new ones
np.setdiff1d(avalanche_com, aval_gdf_com)
```

```
[18]: array(['', 'macot la plagne', 'mont de lans', 'saint bon tarentaise'],
    dtype=object)
```

```
[19]: # replacing old names based on wikipedia search of current names
# https://fr.wikipedia.org/wiki/M%C3%A2cot-la-Plagne
# https://fr.wikipedia.org/wiki/Mont-de-Lans
# https://fr.wikipedia.org/wiki/Saint-Bon-Tarentaise
avalanche.loc[avalanche['commune']=='macot la plagne', 'commune']='la plagne
    ↪tarentaise'
avalanche.loc[avalanche['commune']=='mont de lans', 'commune']='les deux alpes'
avalanche.loc[avalanche['commune']=='saint bon tarentaise',
    ↪'commune']='courchevel'
# checking blankspace in commune
avalanche[avalanche.commune=='']
```

```
[19]:
```

	code accident	date	heure	département	commune	massif	\
73	1112-65-02	2012-02-18	NaN	65	haute bigorre		
416	1718-65-09	2018-02-23	NaN	65	aure louron		

	site	coordonnées	nzone	départ	altitude	\
73	corneblanque			NaN	NaN	
416	versant sud pic d'aygues cluses	42°52'15.8" / 0°10'0.0"			2570.0	

	inclinaison	... évolution	BRA	qualité	\nneige	cohésion	\nneige	\
73	NaN	... descente	3	sèche			NaN	
416	>45	... ski	montée		3		sèche	

	type	\ndépart	cause	départ	dénivelé	\n(mètres)	\
73	linéaire	accidentelle	soi-même			NaN	
416	NaN		linéaire	naturelle	sérac/corniche		

	largeur	\ncassure	\n(mètres)	épaisseur	\ncassure	max.	\n(cm)	Unnamed: 27
73			NaN				NaN	NaN
416			140.0				NaN	NaN

[2 rows x 28 columns]

```
[20]: # after search of sites inserting missing commune names
avalanche.loc[avalanche['code accident']=='1112-65-02', 'commune']='saint lary_
↳soulan'
avalanche.loc[avalanche['code accident']=='1718-65-09', 'commune']='vielle aure'
```

```
[21]: # another check if all communes from avalanche dataframe are included
aval_gdf = pd.merge(gdf, avalanche, left_on='NOM_COM', right_on='commune',_
↳how='inner')
aval_gdf_com = np.sort(aval_gdf['NOM_COM'].unique())
avalanche_com = np.sort(avalanche['commune'].unique())
# yes, finally all communes are in merged dataframe
np.setdiff1d(avalanche_com, aval_gdf_com)
```

```
[21]: array([], dtype=object)
```

```
[22]: # polygons in geodataframe are created for smaller unit then commune which is_
↳called iris
# in some cases there is more iris units for one commune, therefore connect_
↳polygons based on commune
gdf[gdf.NOM_COM=='chamonix mont blanc']
```

```
[22]:
```

	DEPCOM	NOM_COM	IRIS	DCOMIRIS	\
11330	74056	chamonix mont blanc	0101	740560101	
11331	74056	chamonix mont blanc	0104	740560104	
11332	74056	chamonix mont blanc	0102	740560102	
11333	74056	chamonix mont blanc	0103	740560103	

		NOM_IRIS	TYP_IRIS	\
11330	Les Bossons	Les Moussoux	H	
11331	La Tour Argentiere	Les Praz	H	
11332	Les Pelerins	Le Betty	H	
11333	Chamonix Sud	Bois du Bouchet	H	

	geometry
11330	POLYGON ((996579.250 6540846.600, 996804.700 6...
11331	POLYGON ((996579.250 6540846.600, 996536.450 6...
11332	POLYGON ((999526.400 6542740.300, 999544.100 6...
11333	POLYGON ((999810.800 6544114.200, 999821.900 6...

```
[23]: # creating polygons based on commune
# source 1 https://www.earthdatascience.org/workshops/gis-open-source-python/
↳dissolve-polygons-in-python-geopandas-shapely/
# source 2 https://gis.stackexchange.com/questions/287064/
↳dissolve-causes-no-shapely-geometry-can-be-created-from-null-value-in-geopanda/
↳287065
gdf['geometry'] = gdf.buffer(0.01)
commune_boundary = gdf[['DEPCOM', 'NOM_COM', 'geometry']]
```

```
gdf = commune_boundary.dissolve(by='DEPCOM')
```

```
[24]: # merge of new geodataframe with avalanche records
aval_com = pd.merge(avalanche, gdf, how='left',
↳left_on='commune',right_on='NOM_COM')
```

```
[25]: # counting number of avalanche accidents for each commune
aval_final = (aval_com.groupby(['commune']).size()).sort_values(ascending=False)
# results put into new dataframe
aval_final = aval_final.to_frame().reset_index()
aval_final
```

```
[25]:
```

	commune	0
0	chamonix mont blanc	30
1	val d isere	23
2	tignes	23
3	les belleville	15
4	la clusaz	11
..
190	lozzi	1
191	magland	1
192	metzeral	1
193	mittlach	1
194	abondance	1

```
[195 rows x 2 columns]
```

```
[26]: # renaming column to have clear description
aval_final.rename(columns={0:'count_of_avalanche_accidents'}, inplace=True)
aval_final.head()
```

```
[26]:
```

	commune	count_of_avalanche_accidents
0	chamonix mont blanc	30
1	val d isere	23
2	tignes	23
3	les belleville	15
4	la clusaz	11

```
[27]: # creating final geodataframe to be transformed into graph via bokeh library
aval_gdf_final = pd.merge(gdf, aval_final, how='left',
↳right_on='commune',left_on='NOM_COM')
```

```
[28]: aval_gdf_final.sample(10)
```

```
[28]:
```

	geometry	\
5751	POLYGON ((479432.695 6476530.691, 479432.694 6...	
11130	POLYGON ((181013.190 6840299.199, 181013.190 6...	

```

31431 POLYGON ((550227.390 6930544.299, 550227.390 6...
25830 POLYGON ((706890.608 6515912.791, 706870.630 6...
12584 POLYGON ((474952.808 6299790.194, 474952.807 6...
2829 POLYGON ((852824.310 6927399.198, 852824.310 6...
32870 POLYGON ((701778.703 6968169.510, 701778.704 6...
27227 POLYGON ((486725.895 6245421.209, 486725.896 6...
19470 POLYGON ((762720.297 6893970.491, 762720.296 6...
33743 POLYGON ((605451.100 6321186.590, 605451.099 6...

```

	NOM_COM	commune	count_of_avalanche_accidents
5751		pillac	NaN
11130	saint	sauveur	NaN
31431	saint martin de	boscherville	NaN
25830		beaumont	NaN
12584		demu	NaN
2829		tailly	NaN
32870	croix molineaux		NaN
27227		sentous	NaN
19470		chaumuzy	NaN
33743		le verdier	NaN

```

[29]: aval_gdf_final.fillna({'commune':'No avalanche', 'count_of_avalanche_accidents':
    ↪ 'No avalanche'}, inplace = True)
# source https://towardsdatascience.com/
    ↪ a-complete-guide-to-an-interactive-geographical-map-using-python-f4c5197e23e0

```

```

[30]: aval_gdf_final.info()

```

```

<class 'geopandas.geodataframe.GeoDataFrame'>
Int64Index: 36595 entries, 0 to 36594
Data columns (total 4 columns):
geometry                36595 non-null geometry
NOM_COM                 36595 non-null object
commune                 36595 non-null object
count_of_avalanche_accidents  36595 non-null object
dtypes: geometry(1), object(3)
memory usage: 1.4+ MB

```

```

[31]: # importing bokeh elements
import json
from bokeh.io import show, save, output_notebook
from bokeh.models import (CDSView, ColorBar, ColumnDataSource,
                           CustomJS, CustomJSFilter,
                           GeoJSONDataSource, HoverTool,
                           LinearColorMapper, Slider)
from bokeh.layouts import column, row, widgetbox
from bokeh.palettes import mpl

```

```

from bokeh.plotting import figure, output_file
# input GeoJSON source that contains features for plotting
geosource = GeoJSONDataSource(geojson = aval_gdf_final.to_json())

```

```

[32]: # define color palettes
palette = mpl['Viridis'][6]
palette = palette[::-1] # reverse order of colors so higher values have darker
    ↪ colors

# instantiate LinearColorMapper that linearly maps numbers in a range, into a
    ↪ sequence of colors
# and special grey color for nan values
color_mapper = LinearColorMapper(palette = palette, low = 0, high = 30,
    ↪ nan_color = '#d9d9d9')

# define custom tick labels for color bar starting from 1

tick_labels = {'0': '1', '5': '5', '10': '10', '15': '15', '20': '20', '25': '25',
    ↪ '30': '30'}

# create color bar
color_bar = ColorBar(color_mapper = color_mapper,
                    label_standoff = 8,
                    width = 500, height = 20,
                    border_line_color = None,
                    location = (0,0),
                    orientation = 'horizontal',
                    major_label_overrides = tick_labels)

# create figure object

p = figure(title = 'Number of avalance accidents in commune',
          plot_height = 1400,
          plot_width = 1200,
          toolbar_location = 'below',
          tools = 'pan, wheel_zoom, box_zoom, reset')

p.xgrid.grid_line_color = None
p.ygrid.grid_line_color = None

# add patch renderer to figure
communes = p.patches('xs', 'ys', source = geosource,
                    fill_color = {'field' : 'count_of_avalanche_accidents',
                                'transform' : color_mapper},

                    line_color = 'gray',

```

```

        line_width = 0.20,
        fill_alpha = 1)

# create hover tool
p.add_tools(HoverTool(renderers = [communes],
                        tooltips = [('Commune', '@NOM_COM'),
                                    ('Number of avalanche_
→accidents', '@count_of_avalanche_accidents')]))

p.add_layout(color_bar, 'below')

# final visualization can be seen as html page
output_file("mountains_danger.html")
save(p)

```

[32]: 'C:\\Users\\kamil\\OneDrive\\Plocha\\mountain_project\\mountain_avalanche\\mountains_danger.html'

[]: