MELBOURNE CITY OPEN DATA PLAYGROUND

CLUE Business establishment's location and industry classification

Exploratory Data Analysis

Date	Author/Contributor	Change
17-Nov-2021	Steven Tuften	Initial Draft

ATTRIBUTIONS

Jupyter Notebook derivative of data exploration notebook and d2i_tools.py created by Albert Hon in T2 2021.

Package/Library Imports

In [1]:
```
import os
import time
from urllib.request import urlopen
import json
from datetime import datetime
import numpy as np
import pandas as pd
from sodapy import Socrata
import geopandas
import plotly.express as px
from shapely.geometry import Polygon, Point
from d2i_tools import *
import warnings
warnings.simplefilter("ignore")
```

Constants

In [2]:
```
dataset_id = 'vesm-c7r2' # Melbourne CLUE Business establishment's location and industry classification
geoJSON_Id = 'aia8-ryiq' # Melbourne CLUE Block polygons in GeoJSON format

apptoken = os.environ.get("SODAPY_APPTOKEN") # Anonymous app token
domain = "data.melbourne.vic.gov.au"
client = Socrata(domain, apptoken) # Open Dataset connection
```

```
WARNING:root:Requests made without an app_token will be subject to strict throttling limits.
```

[01] Retrieve dataset Metadata

In [3]:
```
metadata_df = loadClientDatasetsMetadata(client)
print('Selected metadata for the dataset of interest')
metadata_df[metadata_df.id.isin([dataset_id])].T
```

```
Selected metadata for the dataset of interest
```

Out[3]:

	77
name	Business establishment and industry classifica...
id	vesm-c7r2
parent_fxf	[bs7n-5veh]
description	Data collected as part of the City of Melbourn...
data_upd_at	2021-11-02T22:13:48.000Z
pv_last_wk	10
pv_last_mth	67
pv_total	2340
download_count	753
categories	[economy, demographics, transportation]
domain_category	Business
domain_tags	[business, census of land use and employment, ...
domain_metadata	[{'key': 'Quality_Known-Issues', 'value': 'Non...
Quality_What's-included	Full dataset has been included
Quality_Update-frequency	Every two years
Quality_Reliability-level	Reliable and timely
How-to-use_Linked-to	NaN
Data-management_Source-data-update-frequency	Every two years
Quality_Known-Issues	None
How-to-use_Further-information	http://www.melbourne.vic.gov.au/clue
Quality_Data-quality-statement	A team of up to 6 surveyors conducts a field s...

[02] Display first few rows

In [4]:
```
dataresource = client.get_all(dataset_id)
dataset = pd.DataFrame.from_dict(dataresource)
print(f'The shape of dataset is {dataset.shape}.')
print('Below are the first 3 rows of this dataset:')
dataset.head(3).T
```

```
The shape of dataset is (20036, 11).
Below are the first 3 rows of this dataset:
```

Out[4]:

	0	1	2
clue_small_area	Melbourne (CBD)	Melbourne (CBD)	Melbourne (CBD)
location	{'latitude': '-37.82121122', 'needs_recoding':...	{'latitude': '-37.82121122', 'needs_recoding':...	{'latitude': '-37.82121122', 'needs_recoding':...
y_coordinate	-37.82121122	-37.82121122	-37.82121122
census_year	2020	2020	2020
anzsic4_code	0	9511	0
x_coordinate	144.9568736	144.9568736	144.9568736
block_id	1	1	1
anzsic4_description	Vacant Space	Hairdressing and Beauty Services	Vacant Space
property_id	108843	108843	108843
bps_base_id	108843	108843	108843
trading_name	62 Rebecca Walk MELBOURNE VIC 3000	14 Rebecca Walk MELBOURNE VIC 3000	86 Rebecca Walk MELBOURNE VIC 3000

[03] Data Pre-processing

Cast Data types before analysis

In [5]:
```
dataset[['census_year', 'anzsic4_code','block_id']] = dataset[['census_year', 'anzsic4_code','block_id']].astype(int)
dataset[['x_coordinate', 'y_coordinate']] = dataset[['x_coordinate', 'y_coordinate']].astype(float)
dataset = dataset.convert_dtypes() # convert remaining to string
dataset.dtypes
```

Out[5]:
```
clue_small_area         string
location                object
y_coordinate           float64
census_year              Int32
anzsic4_code             Int32
x_coordinate           float64
block_id                 Int32
anzsic4_description     string
property_id             string
bps_base_id             string
trading_name            string
dtype: object
```

Are there any missing values?

In [6]:
```
print(dataset.isnull().sum())
```

```
clue_small_area         0
location               29
y_coordinate           29
census_year             0
anzsic4_code            0
x_coordinate           29
block_id                0
anzsic4_description     0
property_id             0
bps_base_id             0
trading_name            1
dtype: int64
```

In [7]:
```
dataset[dataset['x_coordinate'].isnull()]
```

Out[7]:

	clue_small_area	location	y_coordinate	census_year	anzsic4_code	x_coordinate	block_id	anzsic4_description	property_id	bps_base_id	trading_name
92	Melbourne (CBD)	NaN	NaN	2020	4512	NaN	5	Takeaway Food Services	101345	101345	Kiosk 12, Campbell Arcade MELBOURNE VIC 3000
93	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 3-4, Campbell Arcade MELBOURNE VIC 3000
94	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 8A, Campbell Arcade MELBOURNE VIC 3000
95	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 5, Campbell Arcade MELBOURNE VIC 3000
96	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 9, Campbell Arcade MELBOURNE VIC 3000
97	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 8, Campbell Arcade MELBOURNE VIC 3000
98	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Shop 11, Campbell Arcade MELBOURNE VIC 3000
99	Melbourne (CBD)	NaN	NaN	2020	5910	NaN	5	Internet Service Providers and Web Search Portals	101345	101345	Shop 6-7, Campbell Arcade MELBOURNE VIC 3000
100	Melbourne (CBD)	NaN	NaN	2020	4244	NaN	5	Newspaper and Book Retailing	101345	101345	Shop 10, Campbell Arcade MELBOURNE VIC 3000
101	Melbourne (CBD)	NaN	NaN	2020	4242	NaN	5	Entertainment Media Retailing	101345	101345	Shop 1, Campbell Arcade MELBOURNE VIC 3000
102	Melbourne (CBD)	NaN	NaN	2020	0	NaN	5	Vacant Space	101345	101345	Kiosk 13, Campbell Arcade MELBOURNE VIC 3000
4157	Melbourne (CBD)	NaN	NaN	2020	2630	NaN	34	Electricity Distribution	110942	110942	Substation Opposite 77 Queen Street, MELBOURNE...
6292	Melbourne (CBD)	NaN	NaN	2020	7530	NaN	45	Local Government Administration	111492	111492	Melbourne Visitor Booth Bourke Street MELBOURN...
12254	Carlton	NaN	NaN	2020	2630	NaN	259	Electricity Distribution	111450	111450	Substation Opposite 1 Barry Street, CARLTON VI...
12660	Carlton	NaN	NaN	2020	2620	NaN	273	Electricity Transmission	111010	111010	Substation 5, MacPherson Street CARLTON NORTH ...
13766	North Melbourne	NaN	NaN	2020	0	NaN	380	Vacant Space	103467	103467	254 Errol Street NORTH MELBOURNE VIC 3051
13783	West Melbourne (Residential)	NaN	NaN	2020	0	NaN	403	Vacant Space	109755	109755	487-491 Victoria Street WEST MELBOURNE VIC 3003
14274	West Melbourne (Industrial)	NaN	NaN	2020	2630	NaN	502	Electricity Distribution	111448	111448	Substation Lloyd Street WEST MELBOURNE VIC 3003
14447	Kensington	NaN	NaN	2020	8922	NaN	509	Nature Reserves and Conservation Parks Operation	565481	565481	Bellair Street Reserve Bellair Street KENSINGT...
14647	East Melbourne	NaN	NaN	2020	2630	NaN	604	Electricity Distribution	545681	545681	Substation 36 Opposite 96 Simpson Street, EAST...
14790	East Melbourne	NaN	NaN	2020	7530	NaN	613	Local Government Administration	111447	111447	Building Opposite 172 Powlett Street, EAST MEL...
15544	Southbank	NaN	NaN	2020	2630	NaN	739	Electricity Distribution	567772	567772	Substation 99A Sturt Street SOUTHBANK VIC 3006
17623	Parkville	NaN	NaN	2020	2630	NaN	901	Electricity Distribution	545572	545572	Substation 146 Opposite 29-53 Ievers Street, P...
17916	Parkville	NaN	NaN	2020	5809	NaN	931	Other Telecommunications Services	107423	107423	Antenna Off Brens Drive, PARKVILLE VIC 3052
17950	Parkville	NaN	NaN	2020	5809	NaN	931	Other Telecommunications Services	525698	525698	Antenna Off Brens Drive, PARKVILLE VIC 3052
19067	Docklands	NaN	NaN	2020	9499	NaN	1108	Other Repair and Maintenance n.e.c.	678365	678365	125 Harbour Esplanade DOCKLANDS VIC 3008
19736	North Melbourne	NaN	NaN	2020	2812	NaN	2385	Sewerage and Drainage Services	618476	618476	Pumping Station No.2 330 Macaulay Road KENSING...
19737	North Melbourne	NaN	NaN	2020	2812	NaN	2385	Sewerage and Drainage Services	618527	618527	Pumping Station No.5 Sutton Street NORTH MELBO...
19995	Kensington	NaN	NaN	2020	2812	NaN	2540	Sewerage and Drainage Services	618478	618478	Pumping Station No.1 Smith Street KENSINGTON V...

Drop rows with no latitude or longitude?

We will not be using the latitude and longitude at property level so we can leave these two rows in the dataset.

In [8]:
```
## If we wanted to drop these rows we would use the following two commands.

#dataset = dataset.dropna(axis=0)
#print(dataset.isnull().sum())
```

[04] Analyse data in Aggregate

Count of Business Establishments by CLUE small area

In [9]:
```
groupbyfields = ['clue_small_area']
aggregatebyfields = {'anzsic4_code': ["count"]}

maxByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
maxByBlock.head(10)
```

Out[9]:

	clue_small_area	anzsic4_code
		count
0	Carlton	1346
1	Docklands	1648
2	East Melbourne	645
3	Kensington	566
4	Melbourne (CBD)	11056
5	Melbourne (Remainder)	369
6	North Melbourne	1308
7	Parkville	449
8	Port Melbourne	674
9	South Yarra	63

Count of Business Establishments by ANZSIC4 Code

In [10]:
```
groupbyfields = ['anzsic4_description']
aggregatebyfields = {'anzsic4_code': ["count"]}

maxByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
maxByBlock.head(10)
```

Out[10]:

	anzsic4_description	anzsic4_code
		count
0	Accommodation	342
1	Accounting Services	249
2	Adult, Community and Other Education n.e.c.	68
3	Advertising Services	66
4	Aged Care Residential Services	13
5	Air Conditioning and Heating Services	4
6	Air and Space Transport	12
7	Aircraft Manufacturing and Repair Services	1
8	Airport Operations and Other Air Transport Sup...	4
9	Alternative Health Services	23

Count of Business Establishments by ANZSIC4 Code & CLUE small area

In [11]:
```
groupbyfields = ['clue_small_area','anzsic4_description']
aggregatebyfields = {'anzsic4_code': ["count"]}

maxByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
maxByBlock.head(40)
```

Out[11]:

	clue_small_area	anzsic4_description	anzsic4_code
			count
0	Carlton	Accommodation	86
1	Carlton	Accounting Services	22
2	Carlton	Adult, Community and Other Education n.e.c.	4
3	Carlton	Advertising Services	3
4	Carlton	Aged Care Residential Services	1
5	Carlton	Alternative Health Services	1
6	Carlton	Amusement Parks and Centres Operation	2
7	Carlton	Antique and Used Goods Retailing	5
8	Carlton	Architectural Services	20
9	Carlton	Arts Education	5
10	Carlton	Automotive Electrical Services	2
11	Carlton	Bakery Product Manufacturing (Non-factory based)	3
12	Carlton	Banking	6
13	Carlton	Book Publishing	1
14	Carlton	Brothel Keeping and Prostitution Services	2
15	Carlton	Building and Other Industrial Cleaning Services	2
16	Carlton	Business and Professional Association Services	5
17	Carlton	Cafes and Restaurants	170
18	Carlton	Car Retailing	1
19	Carlton	Catering Services	1
20	Carlton	Child Care Services	7
21	Carlton	Chiropractic and Osteopathic Services	3
22	Carlton	Clothing Manufacturing	7
23	Carlton	Clothing Retailing	8
24	Carlton	Clothing and Footwear Repair	1
25	Carlton	Clothing and Footwear Wholesaling	2
26	Carlton	Clubs (Hospitality)	1
27	Carlton	Commission-Based Wholesaling	1
28	Carlton	Communications Equipment Manufacturing	1
29	Carlton	Computer System Design and Related Services	12
30	Carlton	Computer and Computer Peripheral Retailing	2
31	Carlton	Concreting Services	1
32	Carlton	Convenience Store	10
33	Carlton	Creative Artists, Musicians, Writers and Perfo...	5
34	Carlton	Credit Union Operation	1
35	Carlton	Defence	1
36	Carlton	Dental Services	9
37	Carlton	Educational Support Services	1
38	Carlton	Electrical Services	2
39	Carlton	Electrical, Electronic and Gas Appliance Retai...	4

Count of Business Establishments by Block Id

In [12]:
```
groupbyfields = ['block_id']
aggregatebyfields = {'anzsic4_code': ["count"]}

maxByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
maxByBlock.head(10)
```

Out[12]:

	block_id	anzsic4_code
		count
0	1	41
1	2	4
2	4	47
3	5	11
4	6	44
5	11	93
6	12	99
7	13	96
8	14	284
9	15	364

Count of Business Establishments by CLUE small area, Block Id and ANZSIC4 Code

In [13]:
```
groupbyfields = ['clue_small_area','block_id','anzsic4_description']
aggregatebyfields = {'anzsic4_code': ["count"]}

maxByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
maxByBlock.head(10)
```

Out[13]:

	clue_small_area	block_id	anzsic4_description	anzsic4_code
				count
0	Carlton	201	Nature Reserves and Conservation Parks Operation	1
1	Carlton	201	Other Goods and Equipment Rental and Hiring n....	1
2	Carlton	202	Accommodation	1
3	Carlton	203	Accommodation	3
4	Carlton	203	Cafes and Restaurants	2
5	Carlton	203	Child Care Services	2
6	Carlton	203	Creative Artists, Musicians, Writers and Perfo...	1
7	Carlton	203	Vacant Space	1
8	Carlton	204	Accommodation	2
9	Carlton	204	Cafes and Restaurants	1

Plot Business Establishments by Location on map

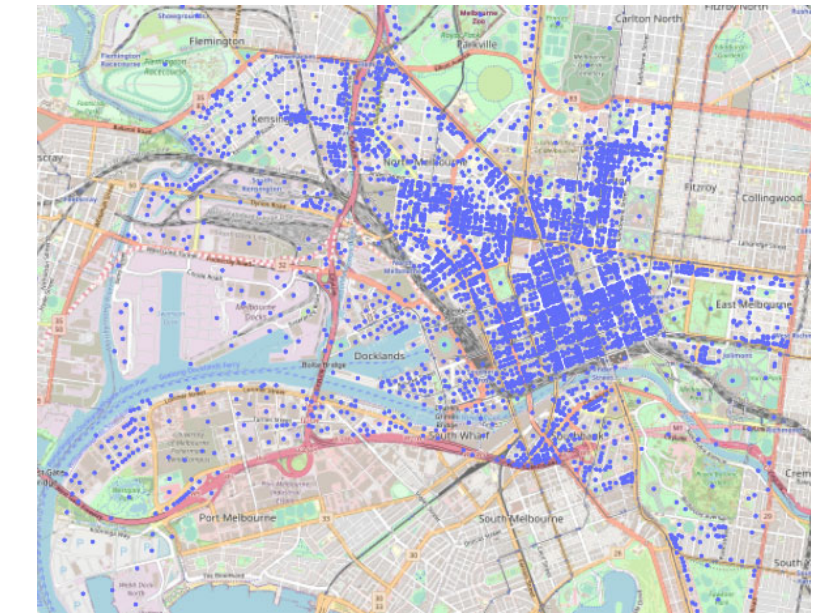
In [14]:
```
groupbyfields = ['clue_small_area','block_id','y_coordinate','x_coordinate']
aggregatebyfields = {'anzsic4_code': ["count"]}

businessesByLocn = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
businessesByLocn.head(10)
```

Out[14]:

	clue_small_area	block_id	y_coordinate	x_coordinate	anzsic4_code
					count
0	Carlton	201	-37.794677	144.965947	1
1	Carlton	201	-37.794367	144.966228	1
2	Carlton	202	-37.794573	144.965299	1
3	Carlton	203	-37.796707	144.965534	1
4	Carlton	203	-37.796680	144.964900	2
5	Carlton	203	-37.796307	144.965281	1
6	Carlton	203	-37.796147	144.965304	1
7	Carlton	203	-37.796069	144.965138	1
8	Carlton	203	-37.796010	144.965758	1
9	Carlton	203	-37.795946	144.965213	1

In [15]:
```
fig = px.scatter_mapbox(businessesByLocn, lat="y_coordinate", lon="x_coordinate",
                        hover_name="clue_small_area",
                        hover_data=["clue_small_area", "block_id"],
                        title='Business Establishments by Location for 2020',
                        zoom=12.5,
                        center = {"lat": -37.813, "lon": 144.945},
                        width=950, height=800)
fig.update_layout(mapbox_style="open-street-map")
fig.show()
```



Plot Dwelling Density by Block

In [16]:
```
groupbyfields = ['block_id','clue_small_area','anzsic4_description']
aggregatebyfields = {'anzsic4_code': ["count"]}

businessesByBlock = pd.DataFrame(dataset.groupby(groupbyfields, as_index=False).agg(aggregatebyfields))
businessesByBlock.columns = businessesByBlock.columns.map(''.join) # flatten column header
businessesByBlock.rename(columns={'clue_small_area': 'clue_area'}, inplace=True) #rename to match GeoJSON extract
businessesByBlock.rename(columns={'anzsic4_codecount': 'business_count'}, inplace=True) #rename to match GeoJSON extract
businessesByBlock.head(10)
```

Out[16]:

	block_id	clue_area	anzsic4_description	business_count
0	1	Melbourne (CBD)	Air and Space Transport	1
1	1	Melbourne (CBD)	Architectural Services	1
2	1	Melbourne (CBD)	Cafes and Restaurants	2
3	1	Melbourne (CBD)	Computer System Design and Related Services	1
4	1	Melbourne (CBD)	Convenience Store	1
5	1	Melbourne (CBD)	Credit Reporting and Debt Collection Services	1
6	1	Melbourne (CBD)	Electricity Distribution	1
7	1	Melbourne (CBD)	General Insurance	1
8	1	Melbourne (CBD)	Hairdressing and Beauty Services	1
9	1	Melbourne (CBD)	Liquor Retailing	1

Get Block Polygon data in GeoJSON format

Load the CLUE Blocks in GeoJSON format and verify the location keys.

In [17]:
```
GeoJSONURL = 'https://'+domain+'/api/geospatial/'+geoJSON_Id+'?method=export&format=GeoJSON'
with urlopen(GeoJSONURL) as response:
    block = json.load(response)

block["features"][0].keys()
block["features"][0]['properties'].keys()
```

Out[17]:
```
dict_keys(['block_id', 'clue_area'])
```

Illustrate Business Establishment Density using a Chloropleth Map using Block regions defined by the GeoJSON data

In [18]:
```
range_max = businessesByBlock['business_count'].max()

fig = px.choropleth_mapbox(businessesByBlock, geojson=block, locations='block_id', color='business_count',
                           color_continuous_scale=["white", "#4444FF", "blue", "darkblue", "#000044"],
                           range_color=(0, 75),
                           featureidkey="properties.block_id",
                           mapbox_style="stamen-toner", #"carto-positron",
                           zoom=12.5,
                           center = {"lat": -37.813, "lon": 144.945},
                           opacity=0.5,
                           hover_name='clue_area',
                           hover_data={'block_id':True,'business_count':True},
                           labels={'business_count':'Number of Businesses','block_id':'CLUE Block Id'},
                           title='Business Establishments by CLUE Block Id for 2020',
                           width=950, height=800
                          )
fig.show()
```

In [ ]:
```
```