Bushifire Risk Score Analysis

Jianing Yu	490056752
Zeyuan Wang	490031276

1. Dataset Description

Most of data sources come from Canvas that teacher give us like csv files and shape files, neighbourhoods dataset describes the census data in Greater Sydney like populations, dwellings and business count.
Statistical areas dataset just give us area name and parent area id.
Businesses stats include many aspects data like assistive services, the accommodation and food, trades, and the agriculture details.
And two shape files, bushfire score in NSW give us the category and the geometry information which can help us to make and see the map, the content has many rows that achieves 510000.
The range of statistical boundary shape file in 2016 from ABS is bigger to use, such as the sa2 code in 2016 or sa2 name, also the geometry is important to use with other tables, we just download to use them.
Probably the bushfire affected the air quality to getting emissions, so the additional dataset we adding is the csv file of air quality, the data comes from OpenDataSoft named World Air Quality-OpenAQ at https://public.opendatasoft.com/explore/dataset/openaq/export/?disjunctive.city&disjunctive.location&disjunctive.measurements_parameter&refine.measurements_sourcename=Australia+-+New+South+Wales, the dataset has collected 231,965,688 air quality measurements from 8,469 locations in 65 countries. Data are aggregated from 105 government level and research-grade sources, however, it's advantageous that filtering the data we want which aims at NSW, there are 236 rows we can use, and we have coordinates, air quality values, pollutant types, timezone and location.
When we uploaded these data, we also did some data cleaning like using dropna functions in SA2 shape files, and fillna function with replacing 0 when there is NULL value, drop.duplication function is also important to delete the same row in dataset, and replace function to drop the comma in numeric value, doing the cleaning can help us in the future calculating.

```
neighbour_data = pd.read_csv('Neighbourhoods.csv')

table_name = "neighbourhoods"
neighbour_data.to_sql(table_name, con=conn, if_exists='replace',index=False)
neighbour_data['land_area'].fillna(0, inplace=True)
conn.execute('''UPDATE neighbourhoods
                    SET population = REPLACE(population,',','')
                ''')
conn.execute('''UPDATE neighbourhoods
                    SET number_of_dwellings = REPLACE(number_of_dwellings,',','')
                ''')
```

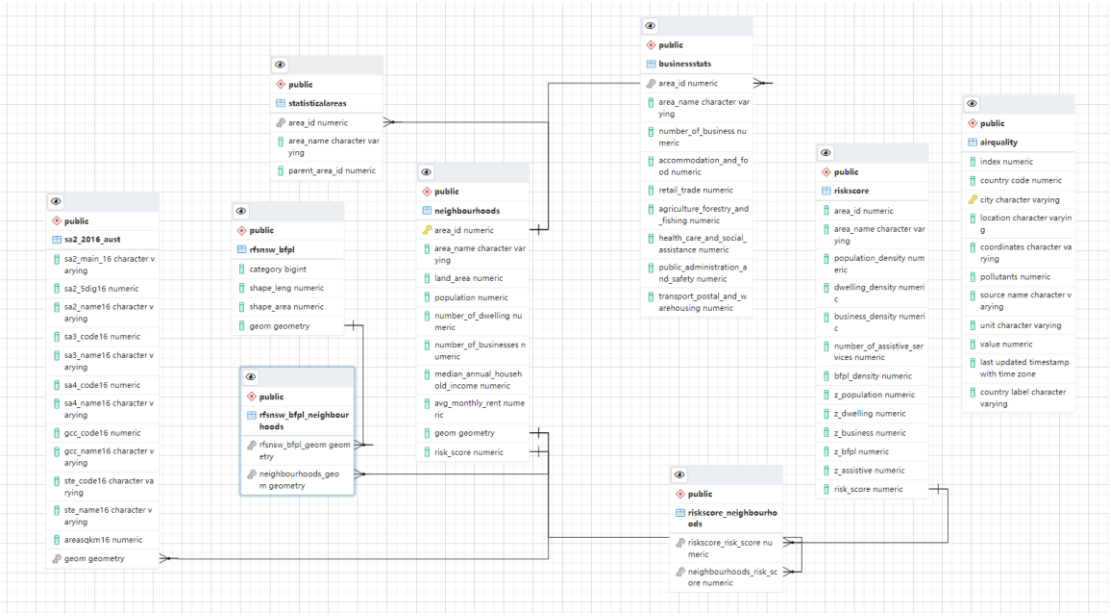
```
]: <sqlalchemy.engine.result.ResultProxy at 0x7f6be4a13290>
```

For shape file, first we need to change all the character in lower case which can fit the data column name, then we need to use dropna function to upload the file.

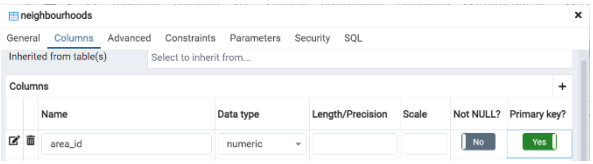
```
sa2WkCpy = sa2.copy()
sa2WkCpy.columns = map(str.lower, sa2WkCpy.columns)
sa2WkCpy['geom'] = sa2['geometry'].dropna().apply(lambda x: create_wkt_element(geom=x, srid=srid))
```

2. Database Description

We uploaded all the dataset into pgAdmin in public schema, we will show database schema by New ERD Project tools.



We use area id in neighbourhoods, businessstats dataset as primary key, because area id can be unique and independent to their dataset, it can distinguish two columns individually. Also, the area id represents each neighbourhood specific and standard area, and this will help us to do queries with other dataset.



Furthermore, we added two foreign key in statisticalarea and sa2 shape file dataset, we use area id and sa2 main code to reference the primary key area id in neighbourhood, this will help us to insert or update the valid entry of data of geometry in sa2 and linking the neighbourhood to calculate the density and fire risk score.

```
sa2_schema = '''CREATE TABLE sa2_2016_aust(
                    sa2_main16 NUMERIC,
                    sa2_5dig16 NUMERIC,
                    sa2_name16 VARCHAR(100),
                    sa3_code16 NUMERIC,
                    sa3_name16 VARCHAR(100),
                    sa4_code16 NUMERIC,
                    sa4_name16 VARCHAR(100),
                    gcc_code16 VARCHAR(100),
                    gcc_name16 VARCHAR(100),
                    ste_code16 NUMERIC,
                    ste_name16 VARCHAR(100),
                    areasqkm16 FLOAT,
                    geom GEOMETRY(MULTIPOLYGON,4283),
                    CONSTRAINT fk_area_id
                    FOREIGN KEY(sa2_main16)
                    REFERENCES neighbourhoods(area_id))
                '''
```

```
stat_schema = """CREATE TABLE IF NOT EXISTS statisticalareas(
                        area_id NUMERIC PRIMARY KEY,
                        area_name VARCHAR(150),
                        parent_area_name VARCHAR(100),
                        CONSTRAINT fk_area_id
                        FOREIGN KEY(area_id)
                        REFERENCES neighbourhoods(area_id))
                        """
```

We joining neighbourhood and sa2 tables together, in query, we just add a column in neighbourhood called geom, then update the sa2 geom column in it, we join them because the sa2 supported the geometry that neighbourhood can use it to link with BFPL dataset to calcultate bfpl density, but the there is no similar area id in BFPL dataset, if we just join neighbourhood with BFPL, it might be messed up. Also, we join the neighbourhoods with risk score table that we store all z-score densities and fire risk score in it. We use area id to limit random 1 value in risk score table to join into neighbourhoods, it helps us to calculate the correlation in future analysis.

```
#Neighbourhoods join statistical area 2 in 2016
conn.execute('''ALTER TABLE neighbourhoods ADD COLUMN geom geometry(MULTIPOLYGON, 4283)''')
conn.execute('''UPDATE neighbourhoods
                    SET geom = (
                        SELECT distinct geom
                          FROM sa2_2016_aust
                         WHERE sa2_2016_aust.sa2_name16 = neighbourhoods.area_name
                    );''')
```

For editing the index value, we thought we will use area id in neighbourhoods frequently, also useful in scanning sa2 shape file to save our time, so we create the index on neighbourhoods area id and index on sa2 main id in sa2_2016_aust, this gives us convenience that we can find the neighbourhood quickly.

```
index_name = "index"
table_name = "neighbourhoods"
column_name = "area_id"

conn.execute('CREATE INDEX {} ON {} ({})'.format(index_name, table_name, column_name))
```

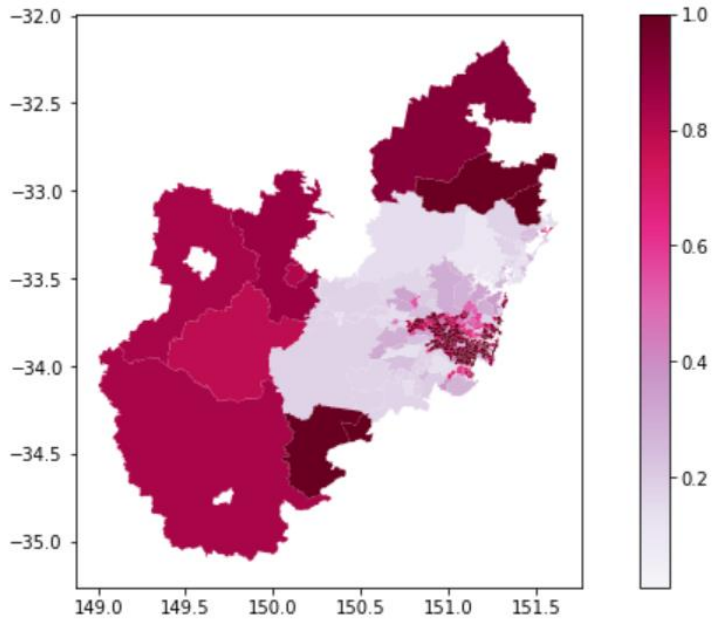
```
<sqlalchemy.engine.result.ResultProxy at 0x7f977817e950>
```

```
index_name = "idx"
table_name = "sa2_2016_aust"
column_name = "sa2_main16"

conn.execute('CREATE INDEX {} ON {} ({})'.format(index_name, table_name, column_name))
```

3. Fire risk score analysis

Indeed, we didn't change the formula teacher gave us, due to limited ability, though we found the additional value of air quality, it's hard to put this two together because the neighbourhoods count is not same, and there is no area_id in our additional file, which can fitted to neighbourhoods, also the unit of the air quality is not same that contains ppm or g^2/m^2. We can support the direct visualization of fire risk score in each neighbourhoods.



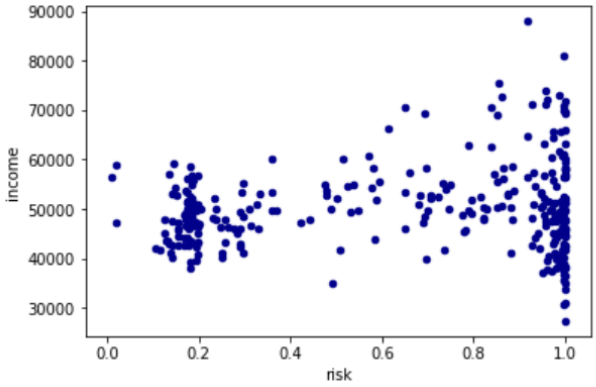
Our map is shown on left side. We change the color of the map that can see the fire risk directly, x-axis and y-axis represent longitude and latitude

of each neighbourhood show in the map, if the color of neighbourhood is deeper, it means the fire risk score is high. From this map, we can catch that if the neighbourhood is closer to the central of city, it will be more safe, I think this related to the city's institutional system is more advanced, the government will be more aim at the system safeguard that most people live around the central, however, the neighbourhood far away from the central will be easy to get fire risk, due to lots of forests, trees and mountains that more closer to nature. Furthermore, we look at the map in Google, we found that there was ocean near the city, we thought maybe this will help that reduce the fire risk and hazard, because the humidity is more enough than neighbourhood which is far away from the ocean.

4. Correlation Analysis

We did both analysis to income and rent with risk score individually.

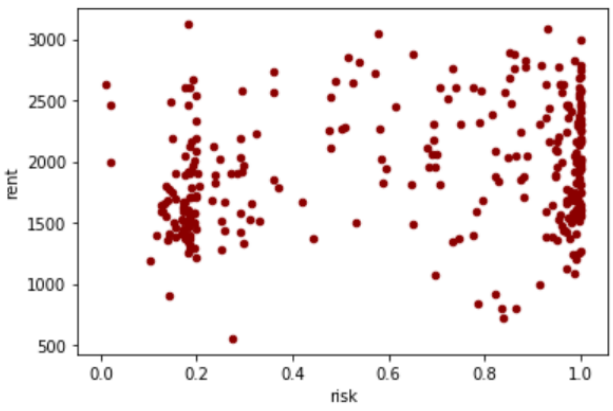
The right scatter plot show the relationship between income and risk, x-axis is risk score, y-axis is the household income, each point represents each neighbourhood. We can roughly see that there is a light correlation between them, whether what the income is, the risk score still increase up in some neighbourhoods.



Then we use the pearson function to calculate the correlation, because pearson describes the linear correlation that taking the values in [-1, 1], and pearson indicates the direction of trend and the degree of trend of change between two variables directly. We use the code below and get the value 0.142.

```
corr1 = correlation['risk'].corr(correlation['income'],method="pearson")
print('pearsonr correlation: %.3f' % corr1)
```

```
pearsonr correlation: 0.142
```



A negative number indicates a negative correlation and a positive number indicates a positive correlation. The higher the absolute value, the stronger the correlation, provided that it is significant. An absolute value of 0 indicates no linear relationship; an absolute value of 1 indicates a full linear

correlation, so this prove our assumption before, they doesn't have strong relationship.

For monthly rent, compare to relationship of income, the distribution distracting more, but this can see is rent is higher, the risk will become higher from the middle part, due to the scatter plot is more regular, we use the spearman function to calculate the correlation The Spearman correlation coefficient is not concerned with whether the two data sets are linearly correlated, but monotonically decreasing or increasing, so the requirement is not much, however, the validity is less than Pearson. First we store the column value in pandas dataframe, which was easy to calculate the results.

```
data2 = pd.read_sql_query("""
        SELECT risk_score, avg_monthly_rent
         FROM neighbourhoods
            """, conn)
correlation2 = pd.DataFrame(
    {'risk': data2['risk_score'],
     'rent': data2['avg_monthly_rent']
    })
```

Just add the calculating method in the last part, it's very convenient.

```
corr2 = correlation2['risk'].corr(correlation2['rent'],method="spearman")
print('spearmanr correlation: %.3f' % corr2)
```

```
spearmanr correlation: 0.232
```

The correlation is 0.232 which represent the rent and risk score have positive relationships, and that's make sense All in all, the fire risk score both correlate to income and rent, this proves our question.

However, since we need to correct more details in code, we thought there are some differences and deviations in risk score calculating, and the last analysis totally depend on these results to see the correlation. We need more exercise to have the big step.