In [12]:

```
%matplotlib  inline
import numpy as np,pandas as pd,matplotlib.pyplot as plt
from sklearn.cluster import DBSCAN
from  geopy.distance import great_circle
from shapely.geometry  import MultiPoint,Polygon
from geopy.geocoders  import Nominatim
from geopy.point  import Point
import geopandas as gpd
from sklearn.preprocessing  import StandardScaler,minmax_scale
import descartes

csv_data = pd.read_csv('airbnb/nylistings.csv')
csv_batch_data = csv_data.tail(5)  # 取后5条数据
print(csv_batch_data)
```

```
             id                          listing_url       scrap
e_id  \
50963  29760869  https://www.airbnb.com/rooms/29760869  2018110315
2857
50964  29760895  https://www.airbnb.com/rooms/29760895  2018110315
2857
50965  29760980  https://www.airbnb.com/rooms/29760980  2018110315
2857
50966  29185060  https://www.airbnb.com/rooms/29185060  2018110315
2857
50967  29761111  https://www.airbnb.com/rooms/29761111  2018110315
2857

      last_scraped                                             nam
e  \
50963   2018-11-04                            Studio in East Harle
m
50964   2018-11-03             NYC Spacious Modern Design Apartmen
t
50965   2018-11-03    Downtown Brooklyn Loft Space Under The Bridg
e
50966   2018-11-04        Large room near JFK - Females/family onl
y
50967   2018-11-04  Clean, convenient, private apt in FLATIRON! :-
)

                                                 summary  \
50963  This is a recently renovated Manhattan studio....
50964  A modern one bedroom one bathroom in a newly b...
50965                                                NaN
50966  Beautiful large room, close to JFK. LIRR, a fi...
50967  This apartment with elevators is perfect for t...

                                                   space  \
50963                                                NaN
50964  This space boasts new hardwood floors, a minim...
```

```
50965                                                NaN
50966  It is a private room (meant only for your use)...
50967  Your private bedroom is one of two rooms in th...

                                             description experienc
es_offered  \
50963  This is a recently renovated Manhattan studio....
none
50964  A modern one bedroom one bathroom in a newly b...
none
50965                                                NaN
none
50966  Beautiful large room, close to JFK. LIRR, a fi...
none
50967  This apartment with elevators is perfect for t...
none

                                neighborhood_overview        ...
\
50963                                             NaN        ...
50964  Restaurants, coffee shops, convenience stores.        ...
50965                                             NaN        ...
50966                                             NaN        ...
50967                                             NaN        ...

      requires_license license jurisdiction_names instant_bookable
\
50963                f     NaN                NaN                t
50964                f     NaN                NaN                t
50965                f     NaN                NaN                t
50966                f     NaN                NaN                t
50967                f     NaN                NaN                f

      is_business_travel_ready         cancellation_policy  \
50963                        f                     flexible
50964                        f  strict_14_with_grace_period
50965                        f                     flexible
50966                        f  strict_14_with_grace_period
50967                        f                     moderate

       require_guest_profile_picture require_guest_phone_verificat
ion  \
50963                             f
f
50964                             f
f
50965                             f
f
50966                             f
f
50967                             f
f
```

```
       calculated_host_listings_count  reviews_per_month
50963                               1                NaN
50964                               3                NaN
50965                               1                NaN
50966                               2                NaN
50967                               1                NaN

[5 rows x 96 columns]
```

```
/Users/ruishang/anaconda3/lib/python3.7/site-packages/IPython/core
/interactiveshell.py:3020: DtypeWarning: Columns (43,87,88) have m
ixed types. Specify dtype option on import or set low_memory=False
.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [55]:

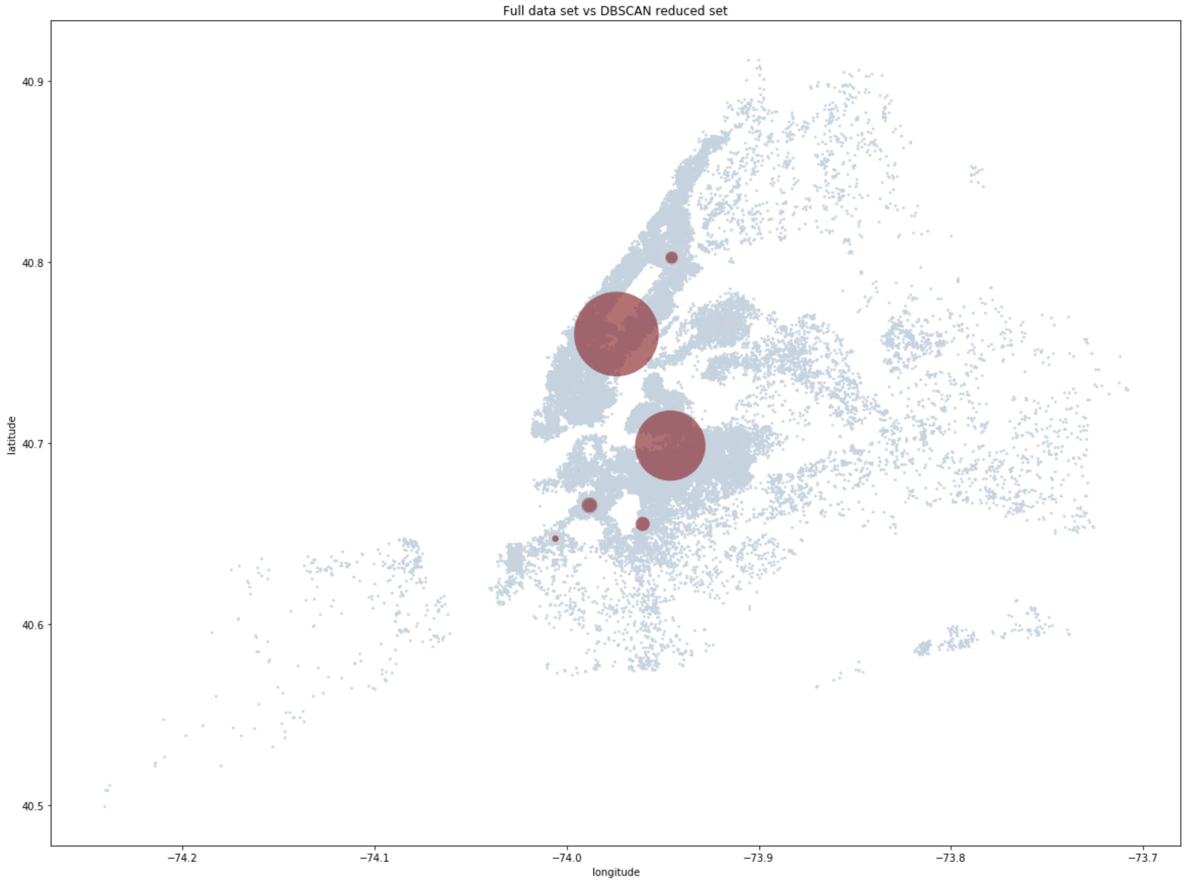
```python
coords=csv_data.as_matrix(columns=['longitude','latitude'])
kms_per_radian = 6371.0088
epsilon = 0.5/ kms_per_radian
db = DBSCAN(eps=epsilon, min_samples=80, algorithm='ball_tree', metric='haversine').fit(np.radians(coords))
cluster_labels = db.labels_
num_clusters = len(set(cluster_labels))
clusters = pd.Series([coords[cluster_labels == n] for n in range(num_clusters)])
print(num_clusters)
# print('Number of clusters: {}'.format(num_clusters)
```

```
/Users/ruishang/anaconda3/lib/python3.7/site-packages/ipykernel_la
uncher.py:1: FutureWarning: Method .as_matrix will be removed in a
future version. Use .values instead.
  """Entry point for launching an IPython kernel.

6
```

In [53]:

```python
def get_centermost_point(cluster):
    centroid = (MultiPoint(cluster).centroid.x, MultiPoint(cluster).centroid.y)
    centermost_point = min(cluster, key=lambda point: great_circle(point, centroid).m)
    return tuple(centermost_point)
centermost_points = clusters[:-1].map(get_centermost_point)

lons, lats = zip(*centermost_points)
rep_points = pd.DataFrame({'lon':lons, 'lat':lats})
rs = rep_points.apply(lambda row: csv_data[(csv_data['latitude']==row['lat']) &(csv_data['longitude']==row['lon'])].iloc[0], axis=1)
rs_latlog = [rs['longitude'], rs['latitude']]
```

In [54]:
```
import folium
import webbrowser
from folium.plugins import HeatMap

scal=minmax_scale([len(line) for line in clusters[:-1]])
fig, ax = plt.subplots(figsize=[20,15])
df_scatter = ax.scatter(csv_data['longitude'], csv_data['latitude']
, c='#C6D3DF', alpha=0.9, s=3)
rs_scatter = ax.scatter(rs['longitude'], rs['latitude'], c='#90353B
', edgecolor='None', alpha=0.7, s=scal*7000)
ax.set_title('Full data set vs DBSCAN reduced set')
ax.set_xlabel('longitude')
ax.set_ylabel('latitude')
plt.show()
```



GMAP plot

In [56]:

```
from matplotlib.cm import viridis
from matplotlib.colors import to_hex
import numpy as np
import gmaps
import gmaps.datasets
import gmaps.geojson_geometries
gmaps.configure(api_key="AIzaSyDlCSfvuDa8cebida7eXgd0-u3BvMwaDaw")
# Your Google API key

# load a Numpy array of (latitude, longitude) pairs


for i in range(1,35):
    rs_lonlat = []
    a = (rs['longitude'][i], rs['latitude'][i])
    rs_lonlat = rs_lonlat.append(a)
    i +=i

array = np.array(a)
```

In [58]:

```
fig = gmaps.figure()
fig.add_layer(gmaps.heatmap_layer(array))
fig
```

```
------------------------------------------------------------------------
---------
AttributeError                            Traceback (most recent c
all last)
~/anaconda3/lib/python3.7/site-packages/gmaps/locations.py in loca
tions_to_list(locations)
     10     try:
---> 11         location_tuples = locations.itertuples()  # locati
ons is a dataframe
     12         locations_as_list = [

AttributeError: 'numpy.ndarray' object has no attribute 'itertuple
s'

During handling of the above exception, another exception occurred
:

TypeError                                 Traceback (most recent c
all last)
<ipython-input-58-227945fd3e6e> in <module>
      1 fig = gmaps.figure()
----> 2 fig.add_layer(gmaps.heatmap_layer(array))
      3 fig

~/anaconda3/lib/python3.7/site-packages/gmaps/heatmap.py in heatma
p_layer(locations, weights, max_intensity, dissipating, point_radi
us, opacity, gradient)
    269         return WeightedHeatmap(**widget_args)
    270     else:
--> 271         return Heatmap(**widget_args)

~/anaconda3/lib/python3.7/site-packages/ipywidgets/widgets/widget.
py in __init__(self, **kwargs)
    409         """Public constructor"""
    410         self._model_id = kwargs.pop('model_id', None)
--> 411         super(Widget, self).__init__(**kwargs)
    412 
    413         Widget._call_widget_constructed(self)

~/anaconda3/lib/python3.7/site-packages/traitlets/traitlets.py in
__init__(self, *args, **kwargs)
    995             for key, value in kwargs.items():
    996                 if self.has_trait(key):
--> 997                     setattr(self, key, value)
    998                 else:
    999                     # passthrough args that don't set trai
ts to super

~/anaconda3/lib/python3.7/site-packages/traitlets/traitlets.py in
__set__(self, obj, value)
    583             raise TraitError('The "%s" trait is read-only.
' % self.name)
    584         else:
--> 585             self.set(obj, value)
    586 
    587     def _validate(self, obj, value):

~/anaconda3/lib/python3.7/site-packages/traitlets/traitlets.py in
set(self, obj, value)
    557 
    558     def set(self, obj, value):
--> 559         new_value = self._validate(obj, value)
    560         try:
    561             old_value = obj._trait_values[self.name]

~/anaconda3/lib/python3.7/site-packages/traitlets/traitlets.py in
_validate(self, obj, value)
    589             return value
    590         if hasattr(self, 'validate'):
--> 591             value = self.validate(obj, value)
    592         if obj._cross_validation_lock is False:
    593             value = self._cross_validate(obj, value)

~/anaconda3/lib/python3.7/site-packages/gmaps/geotraitlets.py in v
alidate(self, obj, value)
     22         if value is None:
     23             return super(LocationArray, self).validate(obj
, value)
---> 24         locations_as_list = locations_to_list(value)
     25         for location in locations_as_list:
     26             latitude, longitude = location
```

```
~/anaconda3/lib/python3.7/site-packages/gmaps/locations.py in loca
tions_to_list(locations)
     17         locations_as_list = [
     18             (latitude, longitude) for (latitude, longitude
)
---> 19             in locations
     20         ]
     21     return locations_as_list

~/anaconda3/lib/python3.7/site-packages/gmaps/locations.py in <lis
tcomp>(.0)
     16     except AttributeError:
     17         locations_as_list = [
---> 18             (latitude, longitude) for (latitude, longitude
)
     19             in locations
     20         ]

TypeError: cannot unpack non-iterable numpy.float64 object
```

In [11]:

```
import gmaps
import gmaps.datasets
gmaps.configure(api_key="AIzaSyDlCSfvuDa8cebida7eXgd0-u3BvMwaDaw")
# Your Google API key

# load a Numpy array of (latitude, longitude) pairs
### https://github.com/pbugnion/gmaps
```

In [2]:

```
fig = gmaps.figure()
fig.add_layer(gmaps.heatmap_layer(locations))
fig
```

In [17]:

```
#Imports required
from shapely.geometry import Point, Polygon
from geopy.distance import great_circle, vincenty
from sklearn.cluster import DBSCAN
from scipy.spatial import ConvexHull
# Spatial clusters based on the histogram
data = csv_data[['latitude', 'longitude']]
db = DBSCAN(eps = 0.0007, min_samples = 8, metric ='euclidean', alg
orithm='auto')
db.fit(data)
# Visualization of clusters with shapely and geojson
coords = csv_data.as_matrix(['latitude', 'longitude'])
cluster_labels = db.labels_
n_clusters = len(set(cluster_labels))
clusters = pd.Series([coords[cluster_labels == n] for n in range(0,
n_clusters)])
maploc1 = folium.Map(tiles='cartodbpositron', location=[51.30, 0.10
],zoom_start=11)
for cluster in clusters:
    if len(np.unique(cluster)) <= 2:
        print ('bad cluster '  + str(cluster))
        continue
    inverted = [[x[1],x[0]] for x in cluster.tolist()]
    ring = Polygon(inverted)
    ring_hull = ring.convex_hull
    folium.GeoJson(mapping(ring_hull)).add_to(maploc1)
    #print(ring)
    #print(mapping(ring_hull))
maploc1
```

```
/Users/ruishang/anaconda3/lib/python3.7/site-packages/ipykernel_la
uncher.py:11: FutureWarning: Method .as_matrix will be removed in
a future version. Use .values instead.
  # This is added back by InteractiveShellApp.init_path()
```

```
------------------------------------------------------------------
---------
AttributeError                            Traceback (most recent c
all last)
~/anaconda3/lib/python3.7/site-packages/shapely/speedups/_speedups
.pyx in shapely.speedups._speedups.geos_linearring_from_py()

AttributeError: 'list' object has no attribute '__array_interface_
_'

During handling of the above exception, another exception occurred
:

ValueError                                Traceback (most recent c
all last)
<ipython-input-17-6624c7c163e5> in <module>
     19         continue
     20     inverted = [[x[1],x[0]] for x in cluster.tolist()]
---> 21     ring = Polygon(inverted)
     22     ring_hull = ring.convex_hull
     23     #folium.GeoJson(mapping(ring_hull)).add_to(maploc1)

~/anaconda3/lib/python3.7/site-packages/shapely/geometry/polygon.p
y in __init__(self, shell, holes)
    238
    239         if shell is not None:
--> 240             ret = geos_polygon_from_py(shell, holes)
    241             if ret is not None:
    242                 self._geom, self._ndim = ret

~/anaconda3/lib/python3.7/site-packages/shapely/geometry/polygon.p
y in geos_polygon_from_py(shell, holes)
    492
    493     if shell is not None:
--> 494         ret = geos_linearring_from_py(shell)
    495         if ret is None:
    496             return None

~/anaconda3/lib/python3.7/site-packages/shapely/speedups/_speedups
.pyx in shapely.speedups._speedups.geos_linearring_from_py()

ValueError: A LinearRing must have at least 3 coordinate tuples
```

In [11]:

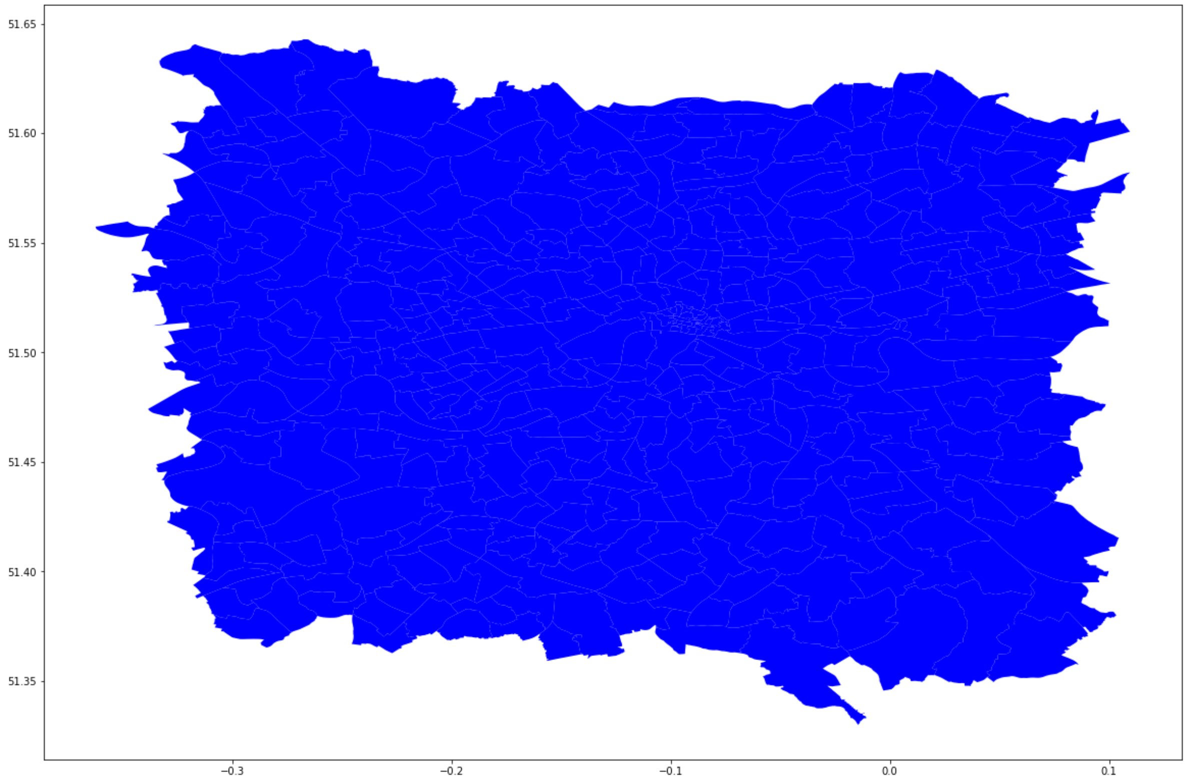
```
margin_width = 0
lon_range = [rs['longitude'].min() - margin_width, rs['longitude'].max() + margin_width]
lat_range = [rs['latitude'].min() - margin_width, rs['latitude'].max() + margin_width]

spatial_extent = Polygon([(lon_range[0], lat_range[0]),
    (lon_range[0], lat_range[1]),
    (lon_range[1], lat_range[1]),
    (lon_range[1], lat_range[0])])
```

In [32]:

```
file_path='shape/London_Ward.shp'
world=gpd.read_file(gpd.datasets.get_path('naturalearth_lowres'))
london_map=gpd.read_file(file_path)
london_map=china_map.to_crs(world.crs)
london_map.plot(color='blue',figsize=(20,20))
```

Out[32]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1a20771128>
```

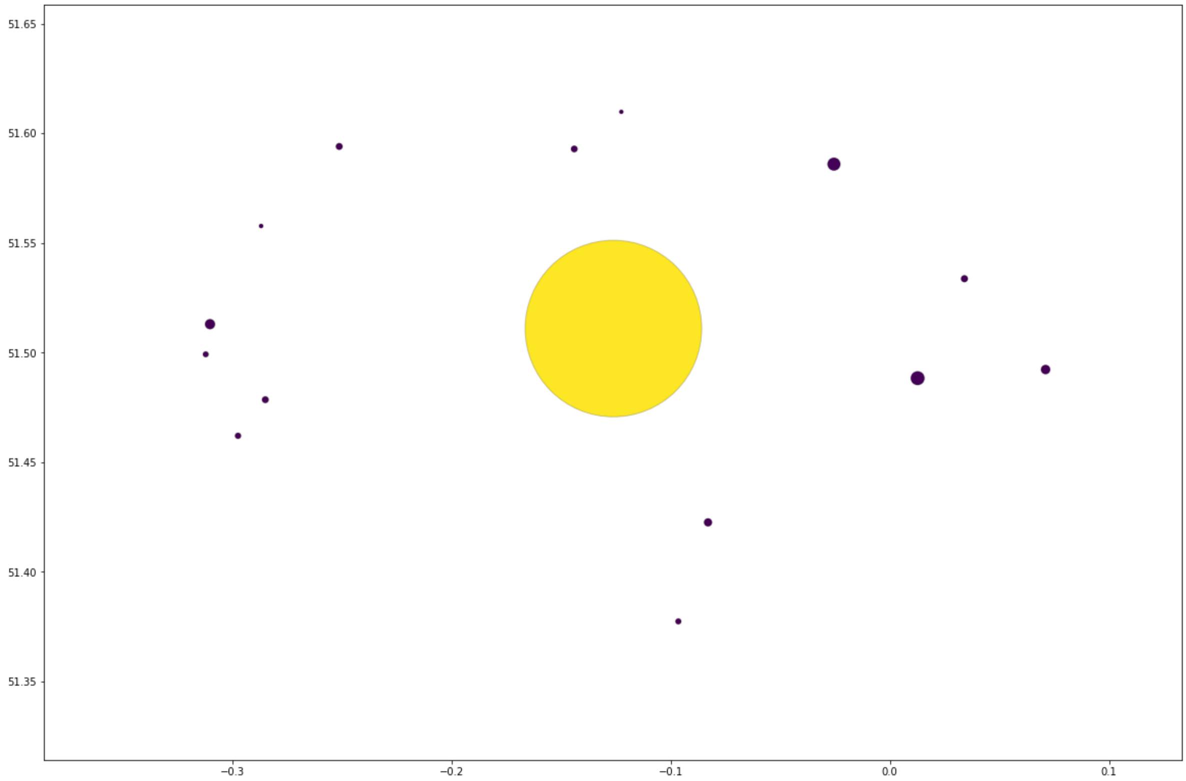


In [36]:

```
import bokeh
from bokeh.plotting import output_file, show

import matplotlib.pylab as pylab
pylab.rcParams['figure.figsize'] = 20, 20
london_map=london_map[london_map['geometry'].intersects(spatial_extent)]
fig=plt.figure()

#set the figure dimensions to the extent of the coordinates in our data
ydimension = int((lat_range[1] - lat_range[0]) / 4)
xdimension = int((lon_range[1] - lon_range[0]) / 4)
fig.set_size_inches(xdimension, ydimension)

# plot the country boundaries and then our point data
china_map.plot(colormap='binary', alpha=0)
scal=minmax_scale([len(line) for line in clusters[:-1]])
rs_scatter = plt.scatter(x=rs['longitude'], y=rs['latitude'], c= scal, edgecolor='black', alpha=2000, s=scal*30000)  # 自己的行为轨迹
```

```
/Users/ruishang/anaconda3/lib/python3.7/site-packages/geopandas/plotting.py:396: FutureWarning: 'colormap' is deprecated, please use 'cmap' instead (for consistency with matplotlib)
  "(for consistency with matplotlib)", FutureWarning)
```

```
<Figure size 0x0 with 0 Axes>
```



In [ ]: