

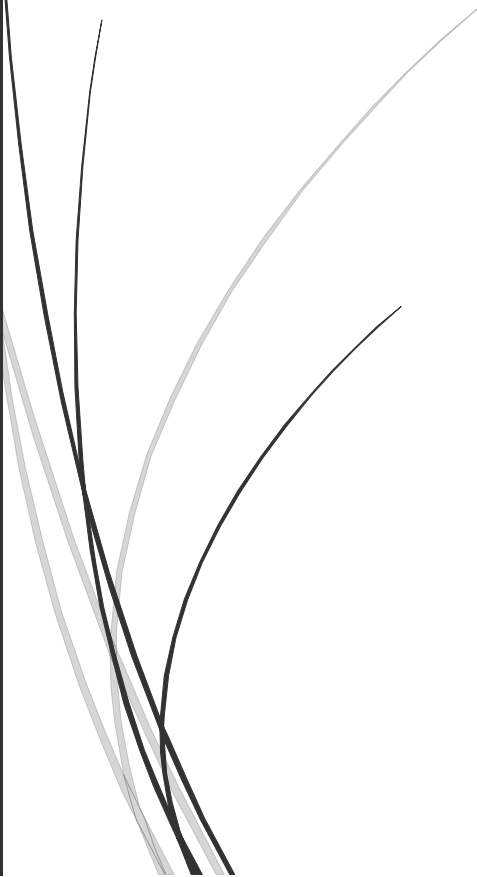
A thick dark grey vertical bar runs down the left side of the page. A red arrow points to the right from the bar, containing the date.

1/29/2021

Applications of Spatial Data Science

Individual Final Report

A spatial analysis and categorical prediction of fuel poverty in London based on socio-economic and urban environment variables

Several thin, curved lines in black and grey originate from the bottom left and sweep upwards and to the right.

Student UG – 1894160

Word Count – 2500

Introduction

Fuel poverty has been a political and socio-economic concern since the study by Isherwood (1979). The concept became a quantifiable measure only in 1991 when Boardman established a 10% income-expenditure threshold, supported by 1988 Family Expenditure Survey for UK households. It remained an official definition until 2011 when John Hills published the 'Low Income High Cost (LIHC)' measure. Current UK fuel poverty policy relies on: 'Households with below average income and higher than average fuel costs are fuel poor'. The changed definition led to 50% decrease of household recognized as fuel poor, due to eradication of low income, low cost and high income, high-cost households (DECC, 2013) and established a division between *poverty* and *fuel poverty* (Middlemiss, 2017).

Consequently, associated indicators, management strategies and aims focus energy efficiency as an origin of the problem (DECC, 2015). Certainly, the role of energy efficiency is essential (Boardman, 2010) and investment in it is a sustainable, cost-effective management approach with long-term environmental benefits (Arimura, 2012). Furthermore, the comparison of the 10% expenditure threshold indicator and the LIHC indicator revealed the superiority of new measure (Robinson, 2018), due to reliance of former on outdated data and oversensitivity to energy prices fluctuations (Moore, 2012). However, overemphasizing efficiency reduced attention to broader causes, ignoring the lived experience of fuel poverty, such as inequality, tenancy problems, ill health, or unstable income (Middlemiss, 2015). Such indirect factors associated with fuel poverty were evaluated by many academics (Romero, 2018; Heidl 2015) and most conclusions are in concordance that socio-demographic and geographic measures play a prominent role in fuel poverty (Morris, 2012; Thomson, 2013). This idea was explored by Besagni (2019), who come up with an alternative, more demographic-focused measures of fuel poverty, and revealed the complex geographical distributions of it in Italy. A similar study of geographic dimensions of fuel poverty has been done in Northern Ireland, where Walker (2012) based the conceptual framework of his research on findings of Liddell (2011): there is a significant spatial correlation between measures of deprivation and fuel poverty. The study identified spatial clusters of a high risk of fuel poverty and argued that a holistic approach with a cross-sector partnership is essential for effective tackling of fuel poverty.

In London, no investigation with geographical approach has been carried out, while most recent 'Fuel Poverty Action Plan' (2018) emphasized that identification and targeting of fuel poor households is difficult with the main challenge being data coverage and exploitation. The latest estimate of 335000 fuel poor households with corresponding £10m management scheme (efficiency-focused) accentuates the importance of further exploration of fuel poverty distribution for potential cost reduction and precise targeting of most vulnerable areas. Thus, based on the drawbacks of LIHC indicator identified in the literature as well as spatial methods not yet applied to London, this investigation intends to provide greater insights into the issue by establishing three aims:

1. To identify fuel poverty predictors unconsidered by LIHC indicator previously
2. To assess the spatial distribution of newly identified and previous independent variables of fuel poverty and detect matches of clusters
3. To construct a classification model to detect areas of most severe fuel poverty

Such a rationale would address the previously unexplored distribution of fuel poverty and its drivers within London. Following the recognized success of Walker's research that established the principle of 'geographic equity' and inspired area-based holistic targeting of fuel poor Irish homes (Baker, 2007), the classification model for London would reveal locations where the issue is felt most severely. Such contribution would provide quantitative guidance to locale investments and with potential to limit expenditure and channel available funds to areas of fuel poverty concentration (Homes, 2011).

Methodology

Variables composing LIHC indicator were reviewed and housing-built period, dwelling type and efficiency rating were selected for further spatial analysis while remaining socio-economic predictors were re-placed with a more holistic, composite measure – Index of Multiple Deprivation (IMD). Most recent data sets were derived from the London Data Store – 2017 recordings of Fuel Poverty and Efficiency and 2015 for remaining variables. All data was available at the LSOA scale, apart from efficiency ratings that were extrapolated to the LSOA level from the borough level to keep the lowest spatial resolution and reduce the MAUP effect (Wong, 2004). Data was converted from strings and integers to floats for further test executions and map plotting, and appropriate columns were inner-merged into single data frame on 'LSOA11CD' column. To accommodate the limited scope of this research, dwelling types and housing built period data was grouped into categories. The newly formed data set was projected to EPSG:27700 and converted to shapefile. Due to the normal distribution of fuel poverty, so a quantile scheme was chosen for further categorization of LSOAs, as this method avoids weaknesses of sparse classes (Brewer, 2002). Finally, the conditional label column was computed using the value ranges of each quantile.

Moran I (Moran, 1948) was chosen as an initial measure of spatial dependence due to its strength in detecting a single dominant type of autocorrelation (Anselin, 2000). To account for variation of spatial dependencies across space, LISA cluster maps were computed to decompose global spatial autocorrelation across space (Anselin, 1995). The combination of two methods was reported as an effective mitigation of LISA's sensitivity to outliers (Tiefelsdorf, 1997). Matches and mismatches of cold and hot spots were analyzed to uncover spatial associations in the discussion section. To quantitatively support the findings, a correlation matrix with Spearman's

Rank method was created. A sample size of 2536 was calculated to be appropriate using Equation 1 (Bonnet, 2000) to avoid false-positive significance from the original large $n = 376002$.

$$N = [(Z\alpha + Z\beta)/C]^2 + 3$$

$Z\alpha = 0.96$ Type I error rate.

$Z\beta = 0.17$ Type II error rate

$$C = 0.5 * \ln[(1+r)/(1-r)] = 0.1003$$

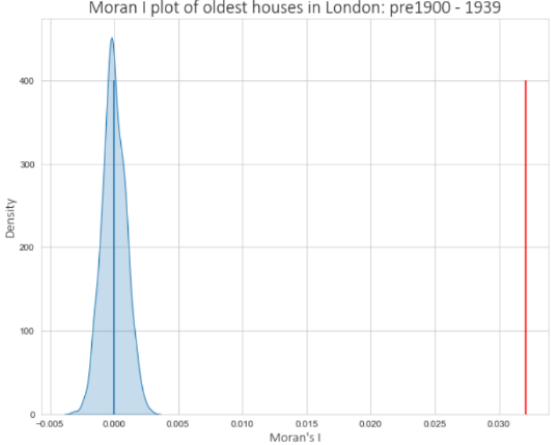
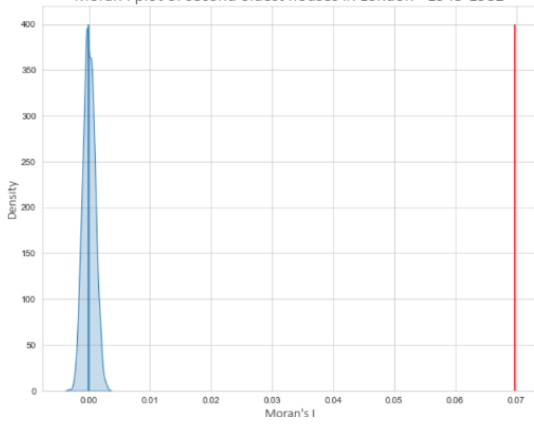
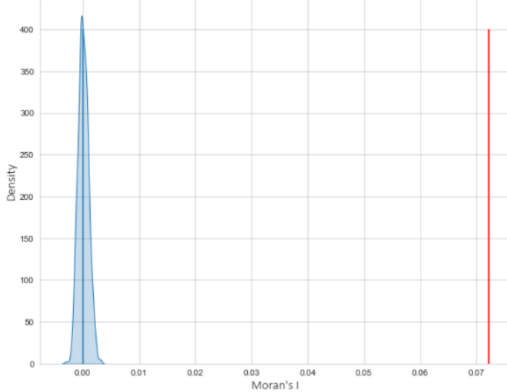
Equation 1: Calculation of sample size

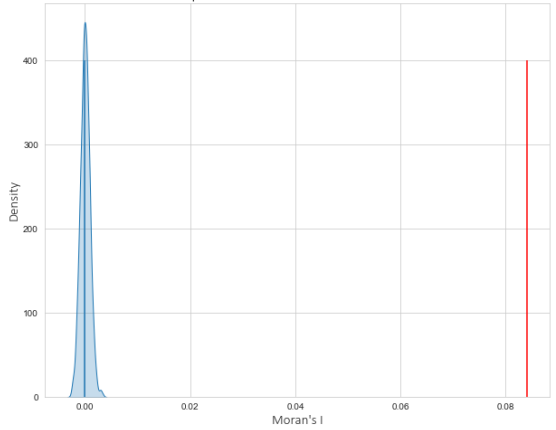
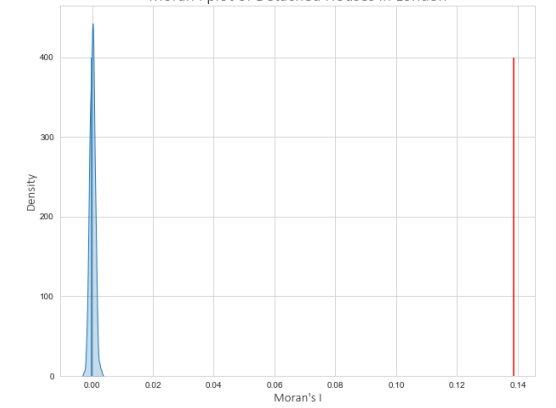
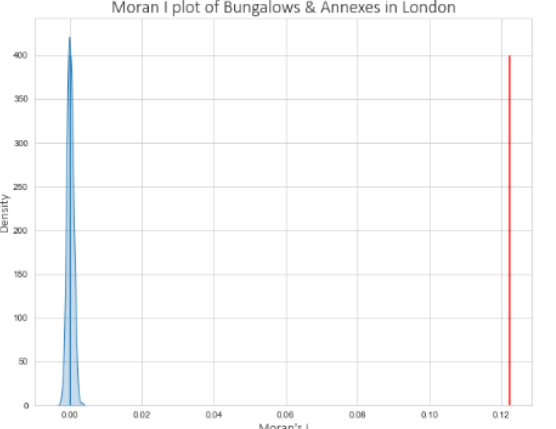
Random Forest Classification algorithm was used to create a model predicting in which quantile an LSOA would fall, following the principles outlined in Walker's (2012) study. Random forest was widely reported to be most robust to multicollinearity, that could handle large data sets (Liu, 2014). It was 39% more accurate than KNN algorithm, as it did not assume dependencies of data points with its surroundings, a recently identified problem in urban studies (Ma, 2020). A total of 15 estimators was fed into a random forest, and 128 trees were used (minimum usage of computational power at highest explanatory power) via manual grid search (Oshiro, 2012). The data was split into 10 cross-validation folds with 80% of data used for training, and 20% for testing, a standard method (Ho, 1995). Strength and reliability of the algorithm were assessed using a confusion matrix and a map of correct and incorrect predictions. Moral I test was performed on errors to check whether they follow a CSR process, corresponding to the unbiased model.

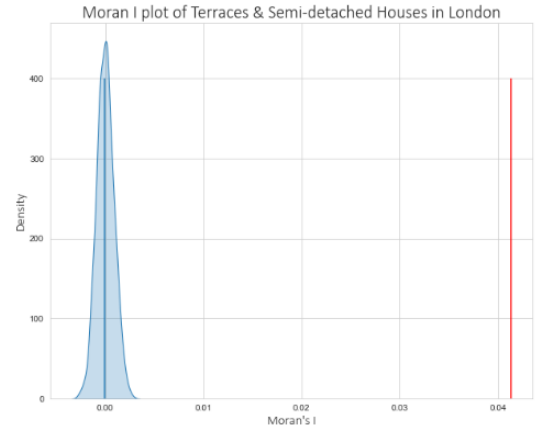
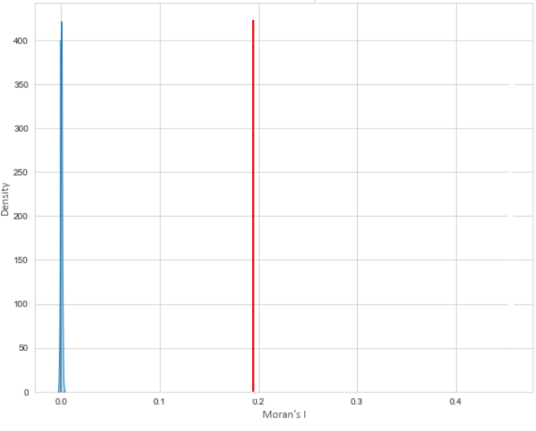
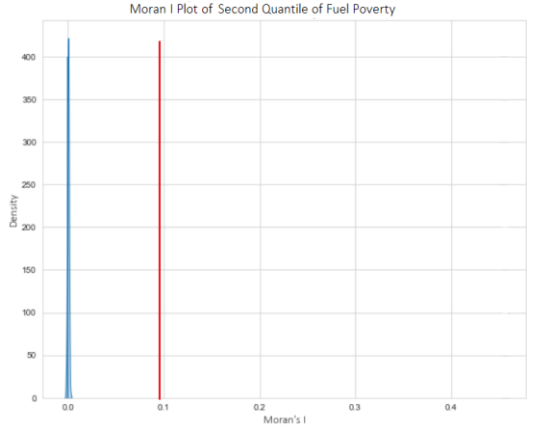
Results

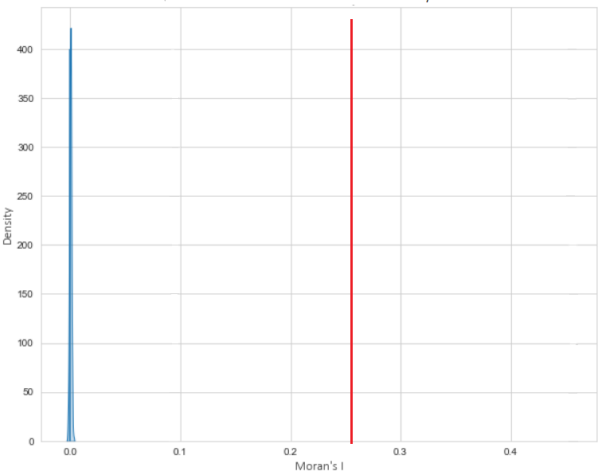
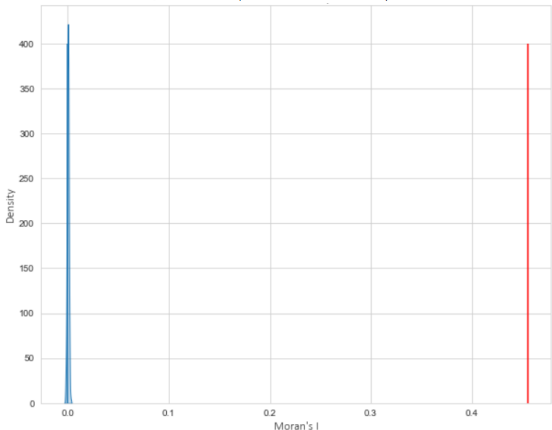
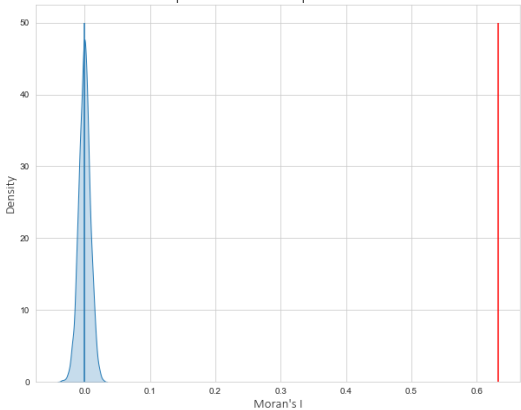
Global Moran I test revealed significant clustering of every variable investigated. Table 1 summarized the data grouping and values of global Moran I test. Most predictors showed minimal degrees of clustering, especially in the housing age category, where values did not exceed 0.072. Most substantial clustering was found in IMD (0.65) and top quantile of fuel poverty (0.47). Efficiency was not examined with Moran I or LISA cluster map, due to borough-level recordings. A proportional stacked bar chart was used to determine the distribution of ratings (see Appendix), showing the negligible amount of best(A) and worst(G) efficiency certificates. Categories B – E were predominant, with B and C taking the highest proportion in every borough.

Table 1: Data Grouping and Results of Moran I test.

Original Variables	Groups used in the study	Moran's I value (0.001 Significance) & Plot
Pre 1900	pre1900-1939(old)	 <p>Moran's I = 0.033</p>
1900 - 1918		
1919 - 1929		
1930 - 1939		
1945 - 1954	1945-1982(medium)	 <p>Moran's I = 0.07</p>
1955 - 1964		
1965 - 1972		
1973 - 1982		
1993 - 1999	1983-2015(new)	 <p>Moran's I = 0.072</p>
1983 - 1992		
2000 - 2009		
2010 - 2015		

FLATS&MAISONETTES	FLATS&MAISONETTES	<p>Moran I plot of Flats & Maisonettes in London</p>  <p>Moran's I = 0.09</p>
DEATCHED	DEATCHED	<p>Moran I plot of Detached Houses in London</p>  <p>Moran's I = 0.139</p>
ANNEXE	BUNGALOW&ANNEXE	<p>Moran I plot of Bungalows & Annexes in London</p>  <p>Moran's I = 0.121</p>
BUNGALOW		

HOUSE_TERRACED	HOUSE_TERRACED&HOUSE_SEMI	 <p>Moran's I = 0.042</p>
HOUSE_SEMI		
Fuel poverty (2017) 2.10 – 38.70%	First Quantile (0 – 9.40)	 <p>Moran's I = 0.19</p>
	Second Quantile (9.40 – 11.40)	 <p>Moran's I = 0.1</p>

	Third Quantile (11.40 – 14.00)	<p>Moran I Plot of Third Quantile of Fuel Poverty</p>  <p>Moran's I = 0.25</p>
	Fourth Quantile (14.00 – 38.70)	<p>Moran I Plot of Top Quantile of Fuel Poverty</p>  <p>Moran's I = 0.47</p>
IMD	IMD	<p>Moran I plot of IMD Score per LSOA in London</p>  <p>Moran's I = 0.65</p>

Lisa cluster maps of all variables were produced and carefully examined (See Appendix). A combined examination of global Moran I test and local distribution of clusters yielded a selected subset of variables with most prominent spatial associations to accommodate the limited extent of this report. The focus was given to:

- Fuel Poverty
- IMD
- Oldest Houses
- Detached Houses

Fuel poverty showed extensive cold spots in South in South-East (Bromley) and West London (Richmond), while hotspots were found in Newham, North Haringey, Merton and on the junction of Brent, Harrow and Ealing.

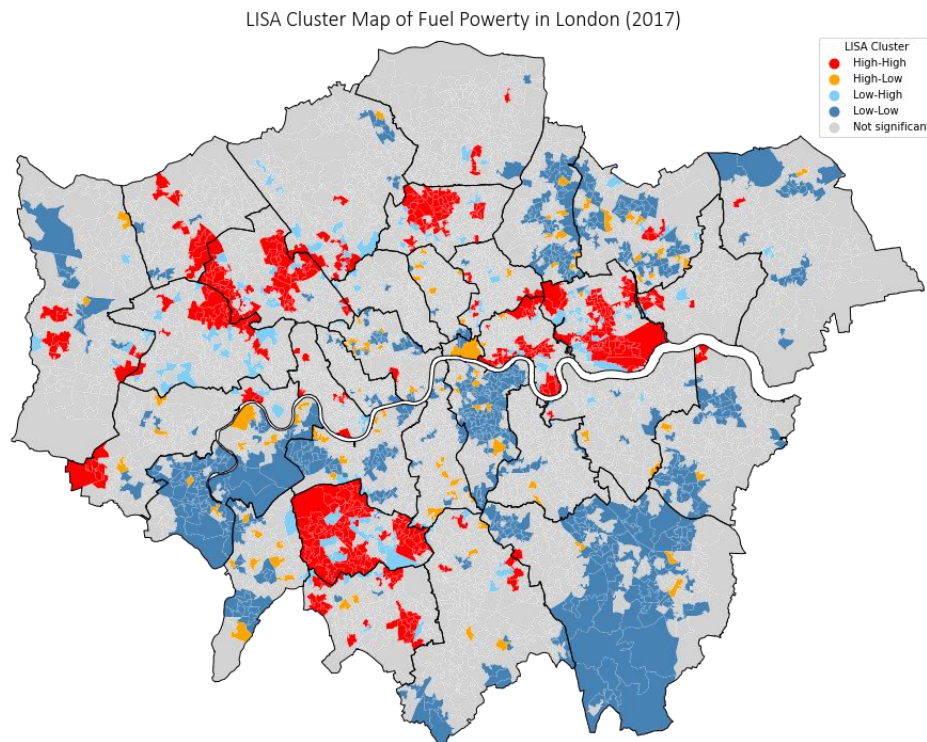


Figure 1: Fuel Poverty clusters (2017)

IMD showed a pronounced trend of hotspots being concentrated predominantly to the North of the River, while cold spots were almost always located closer to city edges. An arc-shaped hotspot curving from Barking and Dagenham to East Enfield is a unique spatial distribution characteristic of IMD, that shows a continuous trend unlike the more borough-dependent distribution of fuel poverty.

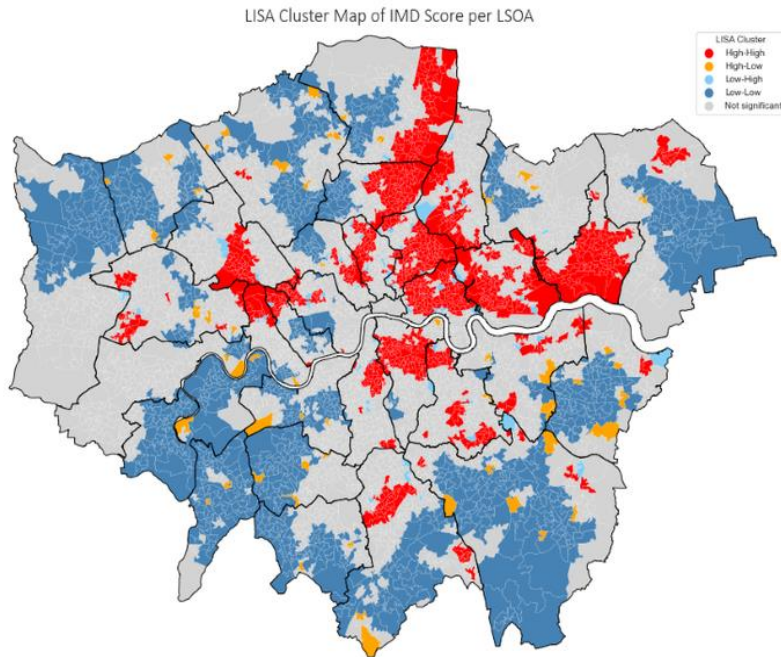


Figure 2: IMD clusters (2015)

Oldest Houses showed the smallest hotspot in Kensington and Chelsea and Westminster with cold spots distributed on the outskirts of London. An inverse trend of significantly higher magnitude showed detached houses data, where extensive cold spot located in the city center spanned to the North-East. Correspondingly, hotspots were distributed around the outer boroughs of London.

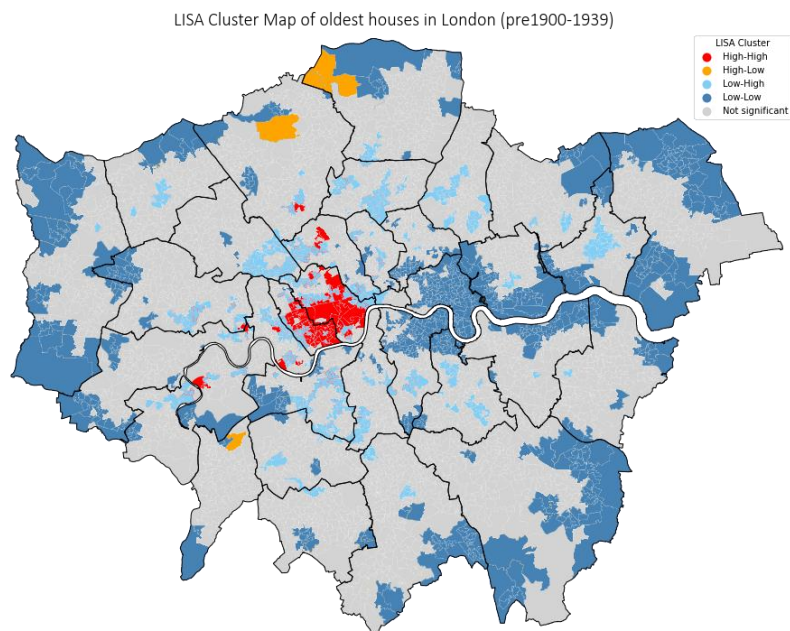
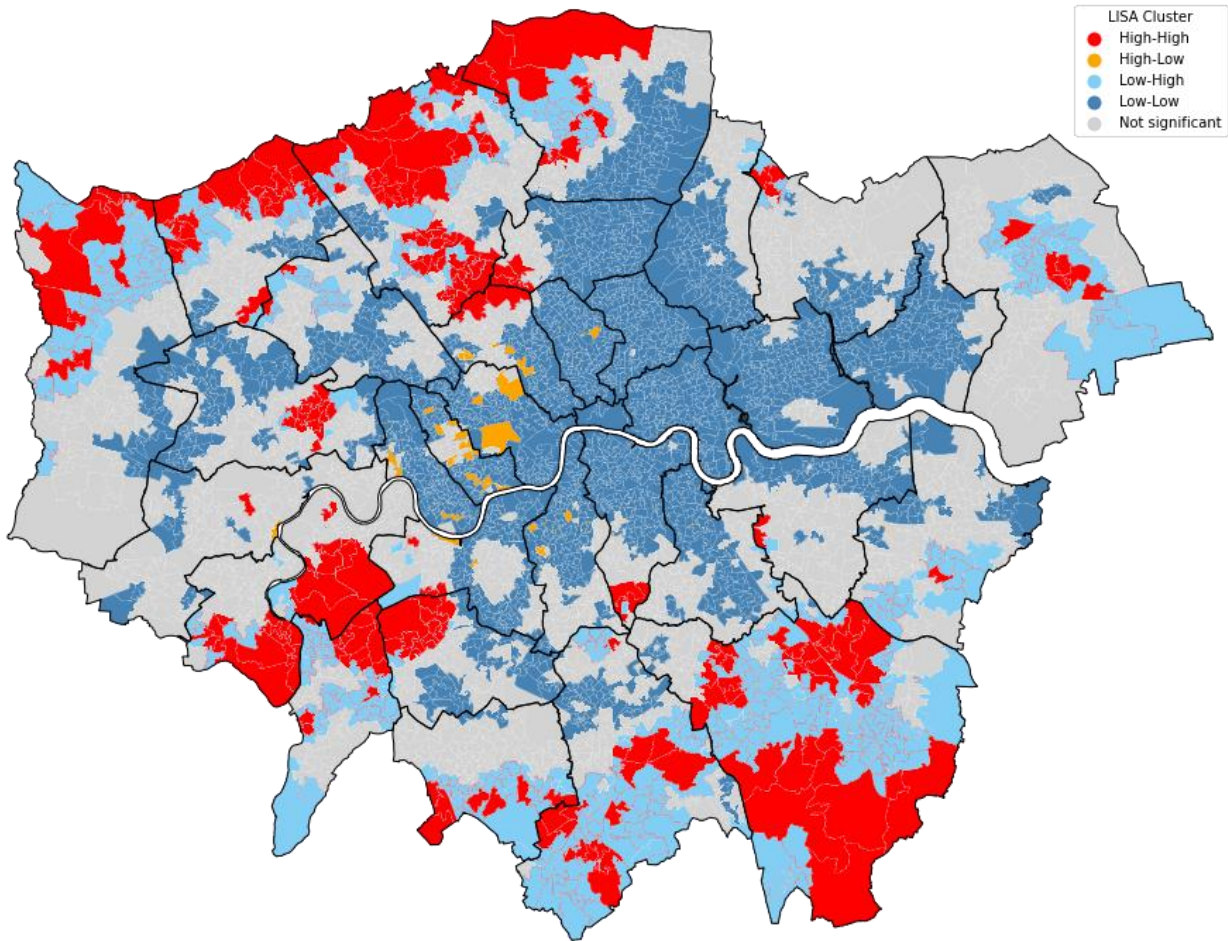


Figure 3: Oldest Dwellings clusters (2015)

LISA Cluster Map of Detached Houses in London (largest dwellings)

*Figure 4: Detached Houses Clusters (2015)*

Overall, it was evident that the cluster matches and mismatches are location-based, and there is no universal trend throughout London. Two case studies of Bromley and Newham will be discussed to evaluate the disparities of spatial associations of variables with fuel poverty across London.

Spearman's rank correlation coefficients, presented in Table 2, were unanimously close to 0, corresponds to uneven relationships across London. For instance, In Newham IMD score correlated more with fuel poverty to the highest extent (0.35), while Richmond experiences weaker relationship (0.04). Yet, the significance of them on 0.05 level supports the presence of influence of predictors of fuel poverty for further step of quantile classification algorithm building.

The overall accuracy of the random forest model was 96% with 0.2% of explanatory power decrease between training and testing data, implying unoverfitted model. The algorithm produced a total of 185 incorrectly classified LSOAs that followed a CSR process (Figure 5), reassuring that there was no bias or significant error. Finally, the confusion matrix revealed a negligible amount of commission and omission errors with even distribution of correct classification per quantile.

Table 2: Spearman's Correlation Coefficients

Variable	Classification	Correlation co-efficient ^a
IMD	IMD	0.27
Housing built period	Old	0.13
	Medium	-0.016
	New	0.012
Dwelling type	Bungalows/Annexes.	-0.036
	Flats and maisonettes.	-0.002
	Terrace and semi-detached.	-0.008
	Detached houses.	0.057
Efficiency	A	-0.021
	B	-0.17
	C	-0.044
	D	-0.098
	E	0.10
	F	0.15
	G	0.21

^aAll are significant at the 0.05 level ($p < 0.05$)

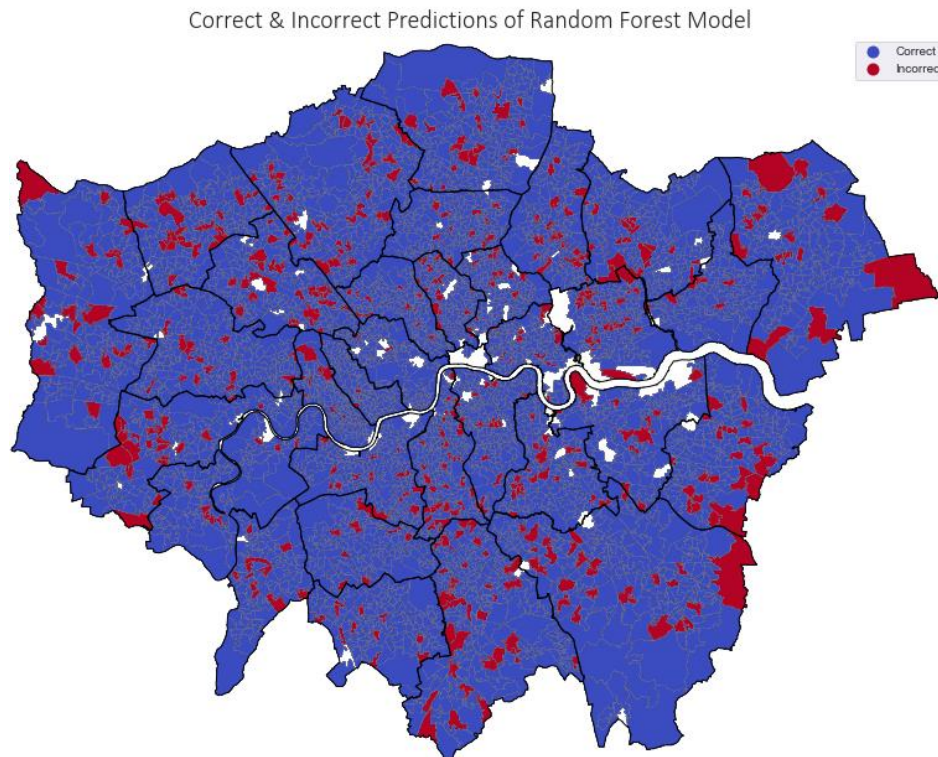


Figure 5: Errors distribution of Random Forest Algorithm

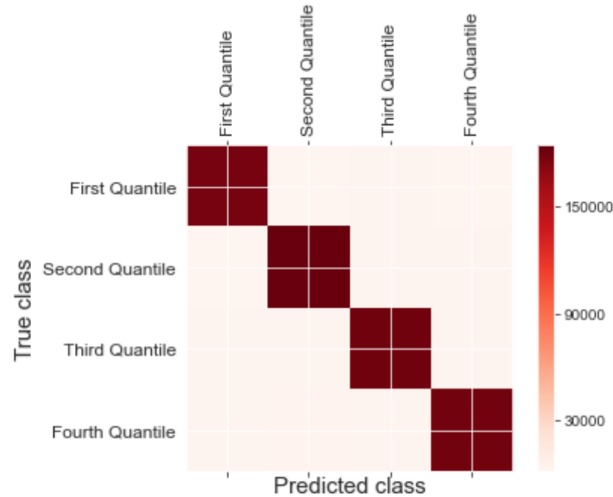


Figure 6: Random Forest Confusion Matrix

Discussion

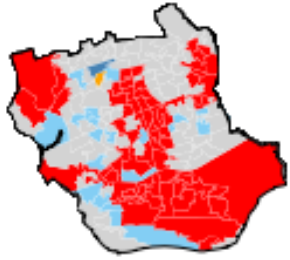
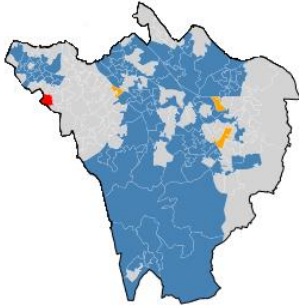
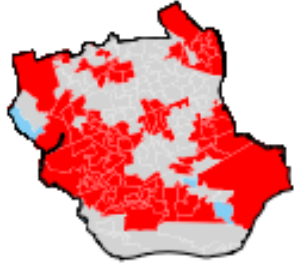
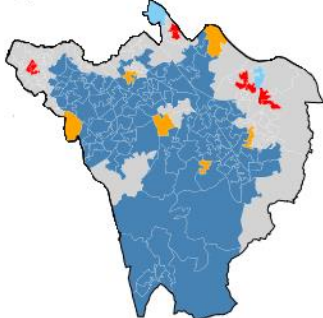
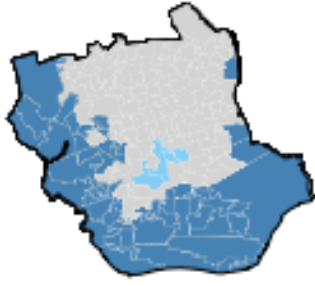

- Main findings & Contributions

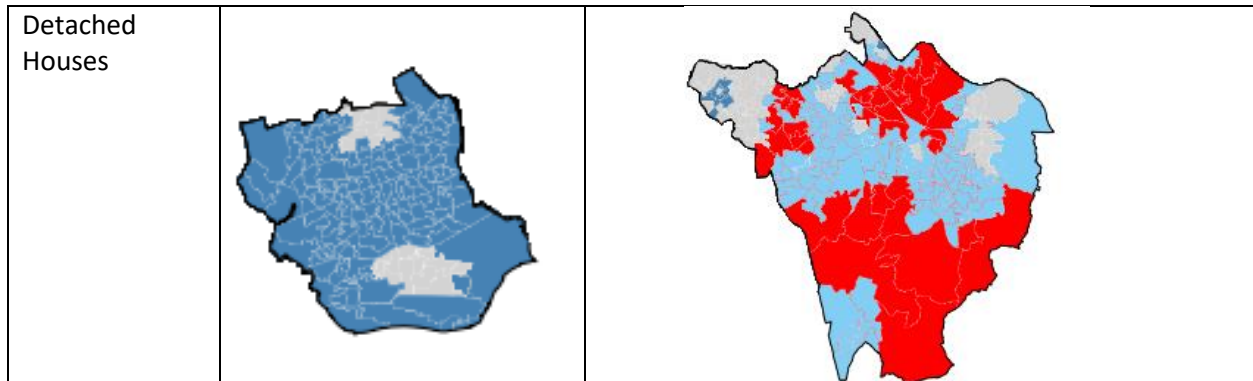
Location-based associations and dependencies of predictors of fuel poverty are complex, and there is no universal relationship in the capital, that is justified by significant yet small correlation coefficients. While (Howden-Chapman, 2012) and (Walker, 2014) found a positive relationship of old dwellings with fuel poverty outside the UK due to low efficiency, such hypothesis had to be rejected for London, as the only hotspot of oldest houses in Kensington did not match with a cluster of fuel poverty. Case studies of Newham and Bromley, chosen due to their spatial fairness and unlike socio-demographic composition (Rugg, 2020) reveal that such association is not evenly distributed across London. Newham fuel poverty cluster matches of oldest houses cold spot and in Bromley only 24% of coldspots LSOA also fall in coldspot of oldest houses. Thus, while we cannot reject the relationship and critique the use of housing age for measurement of fuel poverty as in current LIHC indicator, it is crucial to emphasize that this link is weak. Conversely, detached houses (largest size dwellings) are commonly associated with more well-off socio-economic groups and typically correspond to lower levels of fuel poverty (Roberts, 2015). This relationship is present in both case studies, despite its complexity.

Importantly, IMD has shown one of the strongest and spatially-stable relationships with fuel poverty, illustrated in Bromley and Newham. IMD has been widely used in the UK to aid the identification of places with most severe socio-economic distress and was a successful tool for distribution of investments within thirteen policy projects across ten government departments (ODPM, 2002). Therefore, the main contribution of this research was proposing of IMD - a well-calibrated measure already used for policy initiatives and fund allocation (Deas, 2013) and providing insights into how it could be a strong predictor of fuel poverty, potentially replacing

multiple socio-economic independent variables in the current indicator. Stronger correlation of IMD than Efficiency rating supports the critique of Middlemiss (2017) that efficiency-focused policy approach is a political issue of urban planning and does not necessarily tackle the root of the problem.

Table 3: Spatial correlation of coldspots and hotspots in Newham and Bromley

	Newham	Bromley
Fuel Poverty		
IMD		
Oldest Houses		



Furthermore, the classification model identifying quantiles of fuel poverty at 96% certainty with the use of IMD is a valuable insight into current borough-level management approach. By focusing on the LSOAs in top quantile, looking into their spatial dimensions and recognizing similarities of socio-demographics of those localities, the government could tackle the problem in a novel approach drawing attention to the ambiguities of variables used to measure and define fuel poverty (Healy, 2004). The use of LSOA level data and concentration on London, allowed this study to inspect narrower spatial variations than rural/urban contrast already identified (Roberts, 2015)

- Limitations & Future Work

The temporal lag between fuel poverty (2017) and predictors (2015) is the main limitation of this study along with extrapolated efficiency data set to LSOA level. Thus, we acknowledge that efficiency could play a greater role in fuel poverty than evident from this analysis, yet this would have to be elaborated on using LSOA level recordings. IMD is a composite measure, so indexes associated with it must be further explored to identify only the most valuable and meaningful predictors. Gini Index could be used to determine the predictive strength of each variable fed into Random Forest.

Conclusions

This study explored spatial distributions of variables associated with fuel poverty, identified IMD as potentially a meaningful insight into fuel poverty as well as built a classification model to look to predict areas with highest fuel poverty (top quantile), relying on urban-environmental and socio-economic measures. Thus, the research yielded a better understanding of clusters of fuel poverty and factors leading to them. This geographical approach reveals that borough-level management and investment currently practices mitigating the issue could be adjusted to cluster-level. Emphasis on a socio-demographic and geographic component of fuel poverty is an essential and effective tool for promotion of the knowledge sharing for most effective data-driven strategy design. Area-based programs (such as 'Warm Zones') have shown outstanding effectiveness due to systematic tackling of fuel poor households (Rugkasa, 2007). London has not yet exploited these findings even at an analysis level. Given the responsibility of fuel poverty for

14000 excess winter death in London (ONS, 2017) importance of this research could not be underestimated.

Contributions

Primary responsibilities were processing, cleaning data, writing code, and executing the statistical test. I have created every figure used for this project (Moran I, LISA maps), built a random forest classification model, and wrote a code for the examination of its strengths. Actively engaged in literature review and identified key articles. Collaboratively the team have worked on identification of IMD as a potentially valuable predictor, designed the argument, gathered literature, identified methods for analysis and researched which are our most valuable contributions.

Appendix

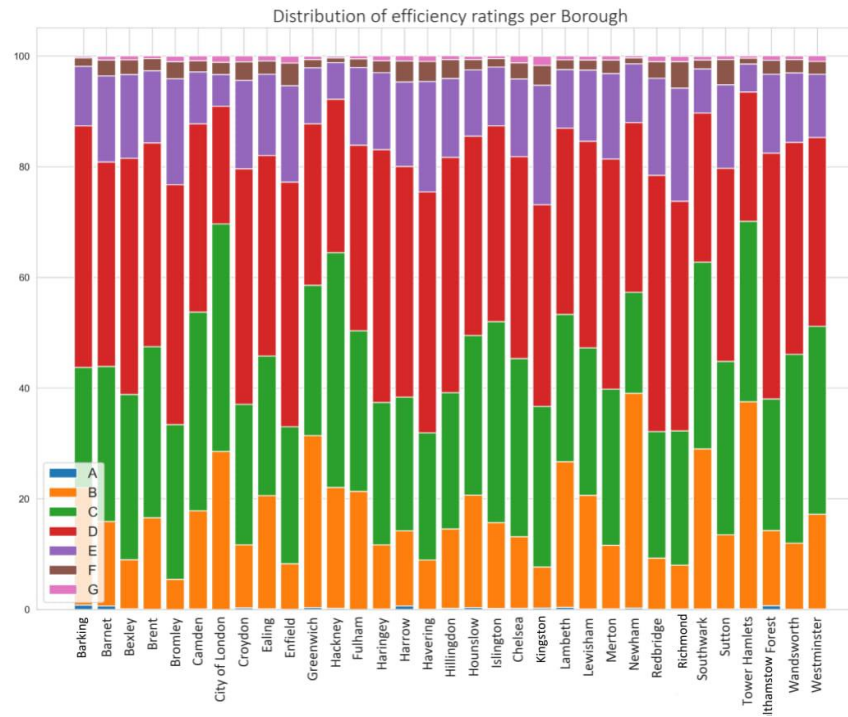


Figure 2: Efficiency Rating Distribution per Borough

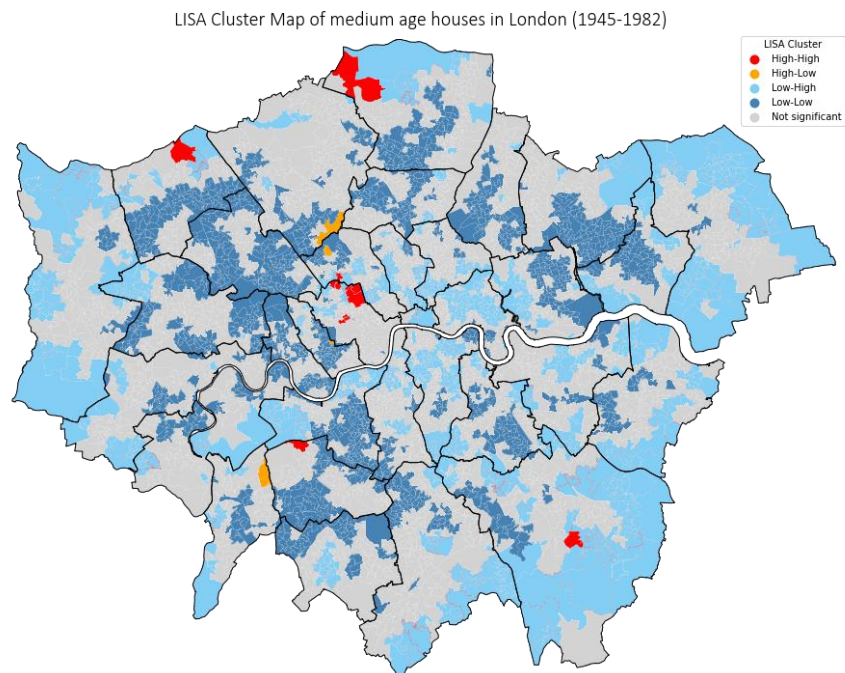


Figure 3: LISA map of medium-age houses

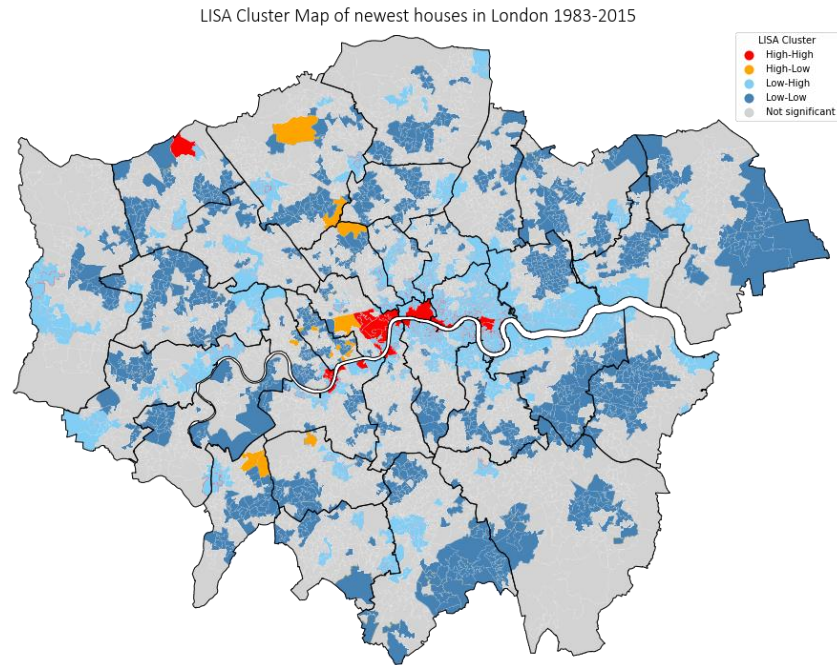


Figure 4: LISA map of newest houses

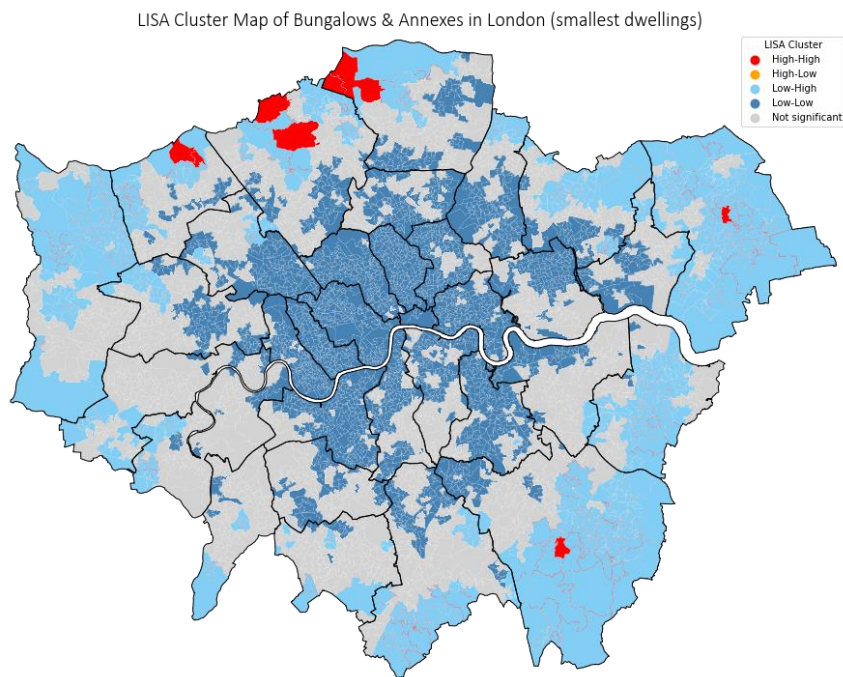


Figure 5: LISA map Of Bungalows and Annexes

LISA Cluster Map of FLats and Maisonettes in London

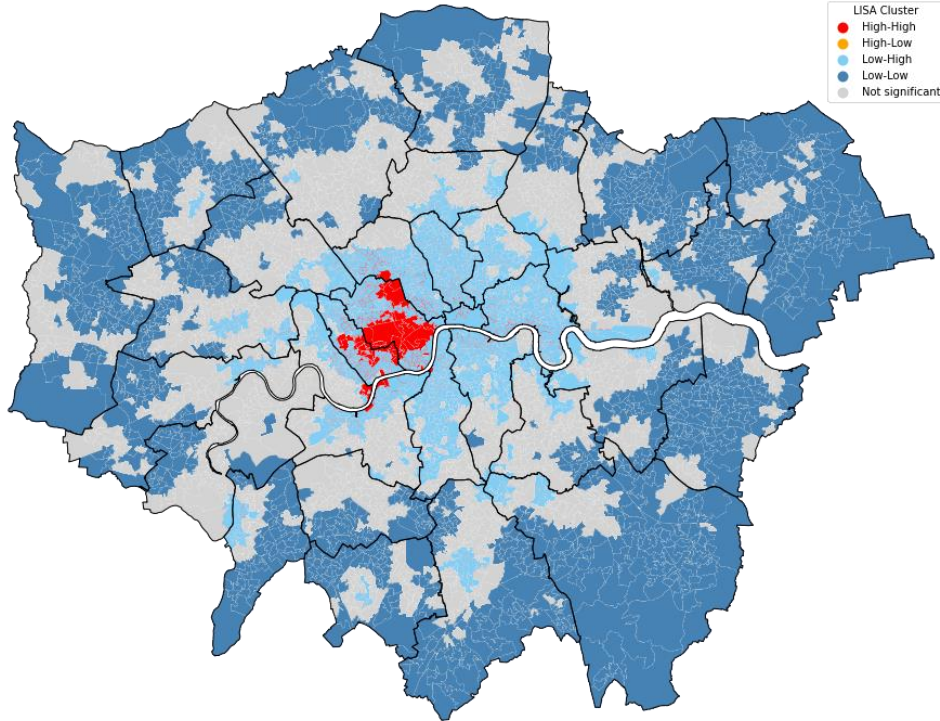


Figure 7: LISA map of flats and Maisonettes

LISA Cluster Map of Terraced and Semi-detached Houses in London

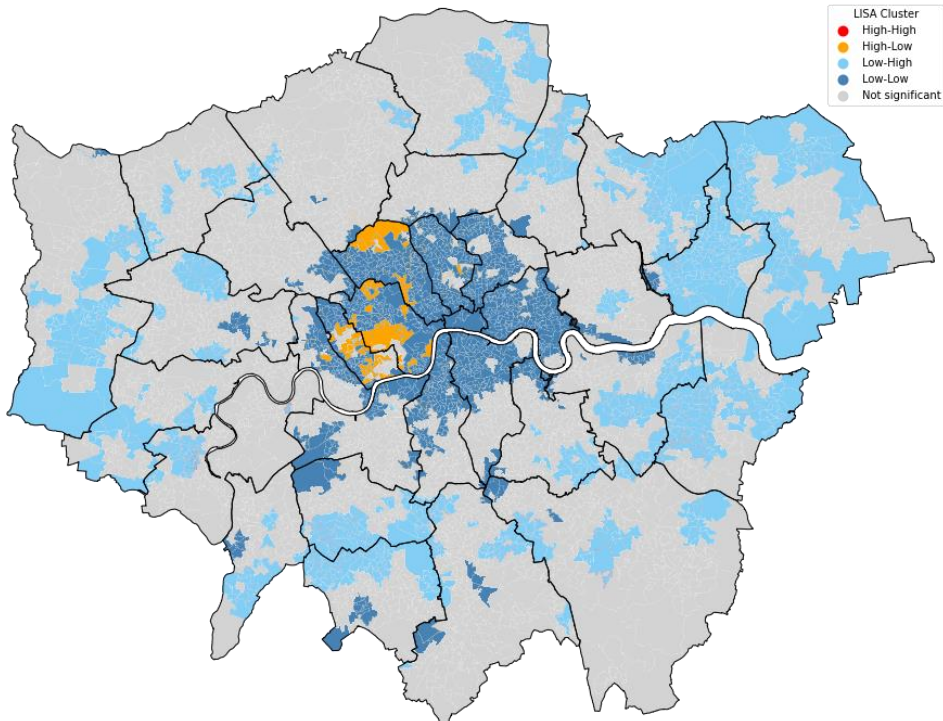


Figure 6: LISA map of Terraced and Semi-detached houses

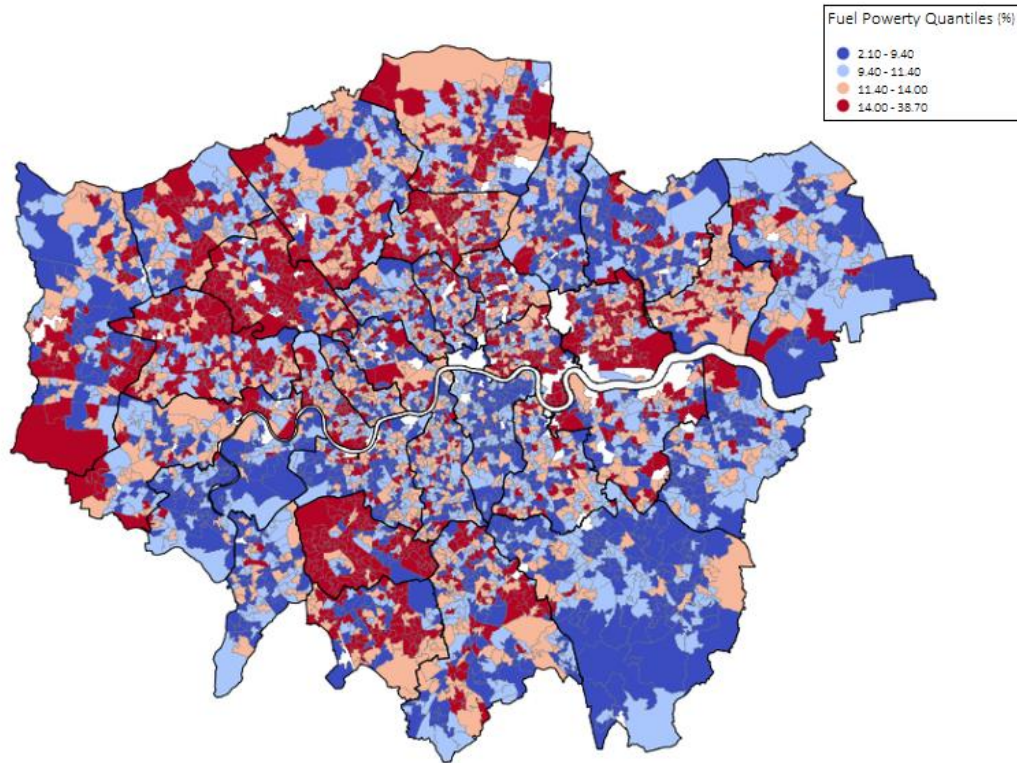


Figure 8: Quantile map of fuel poverty (2017)

Code

```
#Setting up
import os
import urllib
import zipfile
import numpy as np
import pysal as ps
import scipy.spatial as spatial
import geopandas as gpd
from geopandas import GeoDataFrame
import pandas as pd
import shapely.geometry
from shapely.geometry import Point
import matplotlib as mpl
import matplotlib.path as path
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import warnings
```

```

warnings.simplefilter('ignore')
from sklearn.datasets import load_iris
from scipy import stats
from scipy.stats import shapiro
from scipy.stats import normaltest
from libpysal.weights.contiguity import Queen
from libpysal import examples
import numpy as np
import os
import plot
from pysal.explore import esda

#Reading csv data sets
lsoa = gpd.read_file('LSOA_IMD\LSOA_IMD2019.shp', low_memory=False)
lsoa = lsoa[['lsoa11cd', 'geometry']]

fuel_2017 = pd.read_csv('data\fuelpoverty2017.csv', low_memory=False)

imd = pd.read_csv('data\ID_2015_for_London.csv', low_memory=False)

built_period = pd.read_csv('data\dwelling-period-built-2015-lsoa-
msoa.csv', low_memory=False)

dwelling_type = pd.read_csv('data\dwelling-property-type-2015.csv', low_memory=False)
imd = pd.read_csv('data\ID 2015 for London (1).csv', low_memory=False)

efficiency = pd.read_csv('data\energy_performance_2017.csv', low_memory=False)

#Converting data value type from string and integers to floats for each data set and replacing
special characters
for i in fuel_2017[['2012', '2013', '2014', '2015', '2016', '2017']]:
    fuel_2017[i] = fuel_2017[i].replace("-", '0').replace(",", "").astype(float)

imd['IMD Score'] = imd['IMD Score'].replace("-", '0').replace(",", "").astype(float)

for i in built_period[['BP_PRE_1900', 'BP_1900_1918', 'BP_1919_1929',
    'BP_1930_1939', 'BP_1945_1954', 'BP_1955_1964', 'BP_1965_1972',
    'BP_1973_1982', 'BP_1983_1992', 'BP_1993_1999', 'BP_2000_2009',
    'BP_2010_2015', 'BP_UNKNOWN', 'ALL_PROPERTIES']]:
    built_period[i] = built_period[i].replace("-", '0').str.replace(",", "").astype(int)

for i in dwelling_type[['BUNGALOW', 'FLAT_MAIS', 'HOUSE_TERRACED',
    'HOUSE_SEMI', 'HOUSE_DETACHED', 'ANNEXE', 'ALL_PROPERTIES']]:

```

```

dwelling_type[i] = dwelling_type[i].replace("-", '0').str.replace(",", "").astype(float)

for i in efficiency[['A', 'B', 'C', 'D', 'E', 'F', 'G']]:
    efficiency[i] = efficiency[i].replace("-", '0').replace(",", "").astype(float)

#Merging appropriate categories in the data sets
built_period['pre1900-1939(old)'] = built_period['BP_PRE_1900']
+ built_period['BP_1900_1918'] + built_period['BP_1919_1929']
+ built_period['BP_1930_1939']

built_period['1945-1982(medium)'] = built_period['BP_1945_1954']
+ built_period['BP_1955_1964'] + built_period['BP_1965_1972']
+ built_period['BP_1973_1982']

built_period['1983-2015(new)'] = built_period['BP_1983_1992'] + built_period['BP_1993_1999']
+ built_period['BP_2000_2009'] + built_period['BP_2010_2015']

dwelling_type['BUNGALOW&ANNEXE'] = dwelling_type['ANNEXE']
+ dwelling_type['BUNGALOW']

dwelling_type['HOUSE_TERRACED&HOUSE_SEMI'] = dwelling_type['HOUSE_TERRACED']
+ dwelling_type['HOUSE_SEMI']

#Merging data sets
df1 = dwelling_type.merge(lsoa, on = 'lsoa11cd', how = 'inner')
df2 = df1.merge(imd, on = 'lsoa11cd', how = 'inner')
df3 = df2.merge(fuel_2011, on = 'lsoa11cd', how = 'inner')
df4 = df3.merge(fuel_2017, on = 'lsoa11cd', how = 'inner')
df5 = df4.merge(built_period, on = 'lsoa11cd', how = 'inner')
df = df5.merge(efficiency, on = 'lsoa11cd', how = 'inner')

#Converting to a geographical data frame for further analysis
gdf = gpd.GeoDataFrame(df)

#Loading gpkg file
import os
os.makedirs('data', exist_ok=True)
b_path = os.path.join('data', 'Boroughs.gpkg')

if not os.path.exists(b_path):
    boroughs =
gpd.read_file('https://github.com/kingsgeocomp/applied_gsa/raw/master/data/Boroughs.gpkg')
boroughs.to_file(b_path, driver='GPKG')

```

```

print("Downloaded Boroughs.gpkg file.")

else:
    boroughs = gpd.read_file(b_path)
    print("Loaded Boroughs.gpkg file.")

def plt_ldn(b=boroughs):
    fig, ax = plt.subplots(1, figsize=(14, 12))
    b.plot(ax=ax, edgecolor= '#000000', facecolor='None', zorder=3)
    ax.set_xlim([502000,563000])
    ax.set_ylim([155000,201500])
    ax.spines['top'].set_visible(False)
    ax.spines['right'].set_visible(False)
    ax.spines['bottom'].set_visible(False)
    ax.spines['left'].set_visible(False)
    return fig, ax

#Stacked bar chart of energy efficiency (2017)
cumval=0
fig = plt.figure(figsize=(14, 10))
for col in efficiency.columns[~efficiency.columns.isin(['LA Name'])]:
    plt.bar(efficiency['LA Name'], efficiency[col], bottom=cumval, label=col)
    cumval = cumval+efficiency[col]

_ = plt.xticks(rotation=90, fontsize = 15, family = 'Calibri')
_ = plt.legend(fontsize=14)
plt.title(label = "Distribution of efficiency ratings per Borough", fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'})

fig.savefig('Efficiency Stacked Bar Chart .png', dpi=300)

#Creating a conditional labels column of the 4 categories (quantiles) of fuel poverty

gdf['Labels'] = ['First Quantile' if x< 9.40 else 'Second Quantile'if 9.40<=x<11.40 else 'Third
Quantile' if 11.40<=x<14.00 else 'Fourth Quantile' for x in gdf['2017']]
gdf.head()

#Checking the amount of data points in each category and the distribution of fuel poverty
variable
print(len(gdf.Labels == 'First Quantile'))
print(len(gdf.Labels == 'Second Quantile'))
print(len(gdf.Labels == 'Third Quantile'))
print(len(gdf.Labels == 'Fourth Quantile'))
sns.kdeplot(gdf['2017'], color='maroon', label = '2017 Fuel Poverty', legend = True)

```

```

#Cluster map of the 4 quantile categories of fuel poverty
fig, ax = plt_lbn()
gdf.plot(column='2017', scheme='quantiles', k=4, legend=True, ax=ax, edgecolor='grey',
linewidth=0.2, cmap = 'coolwarm')
ax.axis('off')
ax.set_title('Fuel Poverty in 2017', fontdict={'fontsize': '20', 'fontweight' : '3', 'family': 'Calibri'}),
#provide a title
ax.annotate('Source: London Data Store (2011)',xy=(0.1, 0.1), xycoords='figure
fraction', horizontalalignment='left', verticalalignment='top', fontsize=12, color='#555555')
#add source info on the image itself
leg = ax.get_legend()
plt.savefig("Fuel Poverty in 2017 (quantiles map)")
plt.show()

#Local Moran I statistics for fuel poverty (2017)
W_queen= Queen.from_dataframe(gdf)
W_queen.transform = 'r'

from pysal.explore import esda
fig = plt.figure(figsize=(10, 8))
mi = esda.moran.Moran(gdf['2017'], W_queen)
print(mi.I) # Moran's I value
print(mi.p_sim) #Inference on global Moran's I

lisa = ps.explore.esda.Moran_Local(gdf['2017'].values, W_queen, permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of Fuel Poverty in London", fontdict={'fontsize': '20', 'fontweight' : '3',
'family': 'Calibri'})

plt.savefig("LISA Cluster Map of Fuel Poverty")

#Local Moran I statistics for IMD (2015)
W_queen= Queen.from_dataframe(gdf)
W_queen.transform = 'r'

from pysal.explore import esda
mi = esda.moran.Moran(gdf['IMD Score'], W_queen)
print(mi.I) # Moran's I value
print(mi.p_sim) #Inference on Moran's I

lisa = ps.explore.esda.Moran_Local(gdf['IMD Score'].values, W_queen, permutations=999)

```



```

import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of IMD in London", fontdict={'fontsize': '20', 'fontweight': '3',
'family': 'Calibri'})
plt.savefig("LISA Cluster Map of IMD")

#Local Moran I statistics for housing built period (pre 1900-1939)
W_queen= Queen.from_dataframe(gdf)
W_queen.transform = 'r' # row-standardize the contiguity weights

from pysal.explore import esda
mi = esda.moran.Moran(gdf['pre1900-1939(old)'], W_queen)
print(mi.I) # Moran's I value
print(mi.p_sim) #Inference on Moran's I

lisa = ps.explore.esda.Moran_Local(gdf['pre1900-1939(old)'].values, W_queen,
permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of oldest houses (pre1900-1939)", fontdict={'fontsize': '20',
'fontweight': '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of oldest houses (pre1900-1939)")

#Local Moran I statistics for housing built period (1945-1982)
from pysal.explore import esda
mi = esda.moran.Moran(gdf['1945-1982(medium)'], W_queen)
print(mi.I) # Moran's I value
print(mi.p_sim) #Inference on Moran's I

lisa = ps.explore.esda.Moran_Local(gdf['1945-1982(medium)'].values, W_queen,
permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of London's medium-age houses (1945-1982)", fontdict={'fontsize':
'20', 'fontweight': '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of medium-age houses (1945-1982)")

#Local Moran I statistics for housing built period (1983-2015)
from pysal.explore import esda
mi = esda.moran.Moran(gdf['1983-2015(new)'], W_queen)
print(mi.I) # Moran's I value

```

```
print(mi.p_sim) #inference on Moran's I
```

```
lisa = ps.explore.esda.Moran_Local(gdf['1983-2015(new)'].values, W_queen,
permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of London's new houses 1983-2015", fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of new houses 1983-2015")
```

```
#Local Moran I statistics for dwelling type (BUNGALOW&ANNEXE) (2015)
W_queen= Queen.from_dataframe(gdf)
W_queen.transform = 'r'
```

```
from pysal.explore import esda
mi = esda.moran.Moran(gdf['BUNGALOW&ANNEXE'], W_queen)
print(mi.I)
print(mi.p_sim) #Inference on Moran's I
```

```
lisa = ps.explore.esda.Moran_Local(gdf['BUNGALOW&ANNEXE'].values, W_queen,
permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of Bungalows and Annexes in London", fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of Bungalows and Annexes in London")
```

```
#Local Moran I statistics for dwelling type (FLAT_MAIS) (2015)
from pysal.explore import esda
mi = esda.moran.Moran(gdf['FLAT_MAIS'], W_queen)
print(mi.I)
print(mi.p_sim) #Inference on Moran's I
```

```
lisa = ps.explore.esda.Moran_Local(gdf['FLAT_MAIS'].values, W_queen, permutations=999)
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of Flats and Maisonette in London", fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of Flats and Maisonette in London ")
```

```
#Local Moran I statistics for dwelling type (HOUSE_TERRACED&HOUSE_SEMI) (2015)
```

```
from pysal.explore import esda
mi = esda.moran.Moran(gdf['HOUSE_TERRACED&HOUSE_SEMI'], W_queen) #
call moran function
print(mi.I)
print(mi.p_sim) #Inference on Moran's I
```

```
lisa = ps.explore.esda.Moran_Local(gdf['HOUSE_TERRACED&HOUSE_SEMI'].values, W_queen,
permutations=999)
```

```
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of Terraced and Semi-detached Houses in
London", fontdict={'fontsize': '20', 'fontweight': '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of Terraced and Semi-detached Houses in London ")
```

```
#Local Moran I statistics for dwelling type (HOUSE_DETACHED) (2015)
```

```
from pysal.explore import esda
mi = esda.moran.Moran(gdf['HOUSE_DETACHED'], W_queen) # call moran function
print(mi.I)
print(mi.p_sim) #Inference on Moran's I
```

```
lisa = ps.explore.esda.Moran_Local(gdf['HOUSE_DETACHED'].values, W_queen,
permutations=999)
```

```
import splot
from splot.esda import lisa_cluster
lisa_cluster(lisa, gdf, figsize=(15, 11))
plt.title("LISA Cluster Map of Detached Houses in London", fontdict={'fontsize': '20',
'fontweight': '3', 'family': 'Calibri'})
plt.savefig("LISA Cluster Map of Detached Houses")
```

```
#Selecting essential columns from a data frame to speed the code execution
```

```
gdf = gdf[['lsoa11cd', 'BUNGALOW&ANNEXE', 'FLAT_MAIS', 'HOUSE_TERRACED&HOUSE_SEMI',
'HOUSE_DETACHED', 'geometry', 'IMD Score', '2017', 'pre1900-1939(old)',
'1945-1982(medium)', '1983-2015(new)', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'Labels' ]]
```

```
#Correlation matrix (based on Spearman's Rank) based on fuel poverty and associated
variables.
```

```
fig = plt.figure(figsize=(20, 20))
corrMatrix = gdf.sample(500).corr(method= 'spearman')
sns.heatmap(corrMatrix, annot=True)
plt.show()
```

```
#Exporting results to csv file for further analysis and visualisation
```

```

corrMatrix.to_csv('group project\correlation.csv')

#K-NN algorithm (experiment)
from sklearn import preprocessing
from sklearn.preprocessing import LabelEncoder

le = LabelEncoder()
labels = le.fit_transform(gdf.Labels) #so which variable needs to be encoded?
labels

from sklearn.model_selection import train_test_split
attributes = gdf[['pre1900-1939(old)', '1945-1982(medium)', '1983-2015(new)', 'BUNGALOW',
'FLAT_MAIS', 'HOUSE_TERRACED', 'HOUSE_SEMI', 'HOUSE_DETACHED', 'ANNEXE', 'IMD Score',
'A', 'B', 'C', 'D', 'E', 'F', 'G']]
attributes = attributes.values
attributes

train_d, test_d, train_lab, test_lab = train_test_split(attributes, labels)
from sklearn.neighbors import KNeighborsClassifier

knn = KNeighborsClassifier(n_neighbors= 6)
knn.fit(train_d, train_lab)
knn.score(attributes, labels)

from sklearn import metrics
predictions = knn.predict(test_d)
predictions
confusion_matrix = metrics.confusion_matrix(test_lab, predictions)

#Confusion matrix based on K-NN algorithm
plt.matshow(confusion_matrix)
plt.title('Confusion matrix')
plt.colorbar()
plt.ylabel('True label')
plt.xlabel('Predicted label')
plt.savefig('CONFUSION MATRIX knn', dpi=300)
plt.show()

from sklearn.metrics import classification_report
print (classification_report(test_lab, predictions))

#Random Forest construction

```

```

feature_columns = ['pre1900-1939(old)', '1945-1982(medium)', '1983-2015(new)',
'BUNGALOW&ANNEXE', 'FLAT_MAIS', 'HOUSE_TERRACED',
'HOUSE_TERRACED&HOUSE_SEMI', 'IMD Score', 'A', 'B', 'C', 'D', 'E', 'F', 'G']
class_column = 'Labels'

data_features = gdf[feature_columns].values
data_classes = gdf[class_column].values

from sklearn.model_selection import train_test_split
from time import time
from sklearn.ensemble import RandomForestClassifier
train_d, test_d, train_lab, test_lab = train_test_split(data_features, data_classes)

t0 = time() # adding a time() function here, so we know how many minutes has been used

#128 number of trees were selected after manual grid search
clf = RandomForestClassifier(n_estimators=128, n_jobs=-1)
clf.fit(train_d, train_lab)

print("done in %0.3fs." % (time() - t0))
print(clf.score(test_d, test_lab))

from sklearn.model_selection import StratifiedShuffleSplit
test_data_size = 0.2

cross_validation = StratifiedShuffleSplit(n_splits=10, test_size=test_data_size, random_state=0)
cross_validation

group_index = 1
for train_index, test_index in cross_validation.split(data_features, data_classes):
    data_features_train, data_classes_train = data_features[train_index], data_classes[train_index]
    data_features_test, data_classes_test = data_features[test_index], data_classes[test_index]
    print('sub group %d' % group_index)
    print('number of training records is: %d' % len(data_features_train))
    print('number of testing records is: %d' % len(data_features_test))
    group_index = group_index + 1

from sklearn.metrics import confusion_matrix, precision_recall_curve, roc_curve, auc
from sklearn.metrics import confusion_matrix
scores = []
confusion_matrices = []
precision_scores = {}
recall_scores = {}

```

```

pr_auc_scores = {}
fpr_scores = {}
tpr_scores = {}
roc_auc_scores = {}

for train_index, test_index in cross_validation.split(data_features, data_classes):
    data_features_train, data_classes_train = data_features[train_index], data_classes[train_index]
    data_features_test, data_classes_test = data_features[test_index], data_classes[test_index]

    clf = RandomForestClassifier(n_estimators=128, n_jobs=-1)
    clf.fit(data_features_train, data_classes_train)

    # calculating and Saving the scores.
    test_score = clf.score(data_features_test, data_classes_test)
    scores.append(test_score)

    # Saving the confusion matrices.
    data_classes_pred = clf.predict(data_features_test)
    cm = confusion_matrix(data_classes_test, data_classes_pred)
    confusion_matrices.append(cm)

    # Calculating kinds of scores for measuring the performance
    if not pr_auc_scores:
        for c in clf.classes_:
            precision_scores[c] = []
            recall_scores[c] = []
            pr_auc_scores[c] = []
            fpr_scores[c] = []
            tpr_scores[c] = []
            roc_auc_scores[c] = []

    for c in clf.classes_:
        prob_index = np.where(clf.classes_ == c)[0][0]

        precision, recall, _ = precision_recall_curve(data_classes_test,
        prob[:, prob_index], pos_label=c)
        precision_scores[c].append(precision)
        recall_scores[c].append(recall)
        pr_auc_scores[c].append(auc(recall, precision))

        fpr, tpr, _ = roc_curve(data_classes_test, prob[:, prob_index], pos_label=c)
        fpr_scores[c].append(fpr)

```

```

    tpr_scores[c].append(tpr)
    roc_auc_scores[c].append(auc(fpr, tpr))

print ('Accuracy mean: ' + str(np.mean(scores)))
print ('Accuracy std: ' + str(np.std(scores)))

#Confusion matrix based on Random Forest algorithm
classes

classes = ['', 'First Quantile', 'Second Quantile', 'Third Quantile', 'Fourth Quantile']
from matplotlib import cm as cmap
first = True
cm = None

for cm_iter in confusion_matrices:
    if first:
        cm = cm_iter.copy()
        first = False
    else:
        cm = cm + cm_iter

fig, ax = plt.subplots()

colorbar = ax.matshow(cm, cmap=cmap.Reds)
fig.colorbar(colorbar, ticks=[30000, 90000, 150000, 210000, 270000, 330000, 390000])

ax.set_xlabel('Predicted class', fontsize=15)
ax.set_ylabel('True class', fontsize=15)

ax.set_xticklabels(classes, rotation=90)
ax.set_yticklabels(classes)

ax.tick_params(labels=12)
plt.savefig('Confusion Matrix', dpi=300)

#Correct and incorrect classifications based on Random Forest algorithm

gdf['data_classes_pred'] = clf.predict(data_features)
print(len(gdf['data_classes_pred']))
print(len(gdf['Labels']))

gdf['data_classes_pred'] == gdf['Labels']
gdf['Correct_False'] = np.where(gdf['data_classes_pred'] == gdf['Labels'], 'Correct', 'Incorrect')
gdf['Correct_False'].unique()

```

```

fig = plt.figure(figsize=(10, 8))
fig, ax = plt.subplots()
gdf.plot(column='Correct_False', legend=True, ax=ax, edgecolor='grey', linewidth=0.2, cmap =
'coolwarm')
ax.axis('off')
ax.set_title('Correct & Incorrect Predictions of Random Forest Model', fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'}), #provide a title
leg = ax.get_legend()
plt.savefig("Correct & Incorrect Predictions of Random Forest Model")
plt.show()

```

```

#Global Moran I statistics for incorrect classifications in Random Forest model
W_queen= Queen.from_dataframe(gdf)
W_queen.transform = 'r'

```

```

from pysal.explore import esda
fig = plt.figure(figsize=(10, 8))
sns.set_style("whitegrid")
mi = esda.moran.Moran(gdf['HOUSE_TERRACED&HOUSE_SEMI'], W_queen)
print(mi.I) # Moran's I value
print(mi.p_sim) #inference on Moran's I

```

```

ax = sns.kdeplot(mi.sim, shade=True)
ax.set_title(label = " Moran I plot of Random Forest Errors (0.0018)", fontdict={'fontsize': '20',
'fontweight' : '3', 'family': 'Calibri'})
plt.vlines(mi.I, 0, 400, color='r')
plt.vlines(mi.El, 0, 400)
plt.xlabel("Moran's I", fontdict={'fontsize': '15', 'fontweight' : '3', 'family': 'Calibri'})
plt.ylabel("Density", fontdict={'fontsize': '15', 'fontweight' : '3', 'family': 'Calibri'})
plt.savefig("Moran I plot of Random Forest Errors.png", format="png")
plt.show()

```


Bibliography

1. Anselin, L. (1995). *Local indicators of spatial association e LISA*. Geographical Analysis, 27(2), 93e115.
2. Arimura, T. H., Li, S., Newell, R. G., & Palmer, K. (2012). *Cost-effectiveness of electricity energy efficiency programs*. The Energy Journal, 33(2).

3. B. Boardman, (2010) *Fuel Poverty: From Cold Homes to Affordable Warmth*. Pinter Pub Limited, 1991. Boardman B Fixing fuel poverty. London: Earthscan.
4. Baker, E., & Beer, A. (2007). *Developing a working model of housing need: applying geographical concepts and techniques to a problem of public policy*. Applied Geography, 27, 165-180.
5. Besagni, G., & Borgarello, M. (2019). *The socio-demographic and geographical dimensions of fuel poverty in Italy*. Energy Research & Social Science, 49, 192-203.
6. Bonett, D. G., & Wright, T. A. (2000). *Sample size requirements for estimating Pearson, Kendall and Spearman correlations*. Psychometrika, 65(1), 23-28.
7. C. Robinson, S. Bouzarovski, S. Lindley, (2018) *Getting the measure of fuel poverty': the geography of fuel poverty indicators in England*, Energy Res. Soc. Sci. 36 79–93.
8. Chalmers C, (2000) *"The 2001 Index of Multiple Deprivation"*, unpublished report to the Greater London
9. Deas, I. (2013). *Towards post-political consensus in urban policy? Localism and the emerging agenda for regeneration under the Cameron government*. Planning Practice & Research, 28(1), 65-82.
10. DECC (2013) *Fuel poverty: A framework for future action*. Available at: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/211180/FuelPov-Framework.pdf
11. DECC (2015) *Cutting the cost of keeping warm: A fuel poverty strategy for England*. Available at: <https://www.gov.uk/government/publications/cutting-the-cost-of-keeping-warm>.
12. Energy Research & Social Science 6: 146–154. Middlemiss, 2017
13. Fuel Poverty Action Plan for London (2018), Published by Greater London Authority, Available Online at: https://www.london.gov.uk/sites/default/files/fuel_poverty_action_plan.pdf, Accessed December 5, 2020
14. Healy, J., Clinch, J., 2004. *Quantifying the severity of fuel poverty, its relationship with poor housing and reasons for non-investment in energy-saving measures in Ireland*. Energy Policy 32, 207–220.
15. Heindl, P. (2015). *Measuring fuel poverty: General considerations and application to German household data*. FinanzArchiv/Public Finance Analysis, 178-215.
16. Hills J (2011) *Fuel poverty: the problem and its measurement. Interim report of the Fuel Poverty*
17. Ho, T. K. (1995, August). *Random decision forests*. In Proceedings of 3rd international conference on document analysis and recognition (Vol. 1, pp. 278-282). IEEE.
18. *household energy vulnerability through the lived experience of the fuel poor*.
19. Howden-Chapman, P., Viggers, H., Chapman, R., O'Sullivan, K., Barnard, L. T., & Lloyd, B. (2012). *Tackling cold housing and fuel poverty in New Zealand: a review of policies, research, and health impacts*. Energy Policy, 49, 134-142.
20. Isherwood, B. C., & Hancock, R. M. (1979). *Household Expenditure on Fuel: Distributional Aspects*, Economic Advisers Office. DHSS, London, United Kingdom.
21. Liddell C (2011) *The missed exam: Conversations with Brenda Boardman*. Energy

22. Liu, Y, Chen, M., Mao, S. (2014). *Big data: A survey*. Mobile networks and applications, 19(2), 171-209. (ODPM, 2002)
23. Ma, J., Cheng, J. C., Jiang, F., Chen, W., & Zhang, J. (2020). *Analyzing driving factors of land values in urban scale based on big data and non-linear machine learning techniques*. Land Use Policy, 94, 104537.
24. Middlemiss L and Gillard R (2015) *Fuel poverty from the bottom-up: Characterising*
25. Middlemiss, L. (2017). *A critical analysis of the new politics of fuel poverty in England*. Critical Social Policy, 37(3), 425-443.
26. Moore, R. (2012). *Definitions of fuel poverty: Implications for policy*. Energy policy, 49, 19-26. Hills 2011
27. Moran, P. A. (1948). *The interpretation of statistical maps*. Journal of the Royal Statistical Society: Series B (Methodological), 10(2), 243-251.
28. Morris, C., & Liddell, C. (2012). *Seasonality of mortality in Northern Ireland*. Belfast: Annual Report of the Registrar General. Thomson, 2013
29. Office for National Statistics (ONS) (2017). *Statistical bulletin: Excess winter mortality in England and Wales: 2016/17 (provisional) and 2015/16 (final)*. [Online]. Available from: <https://www.ons.gov.uk/peoplepopulationandcommunity/birthsdeathsandmarriages/deaths/bulletins/excesswintermortalityinenglandandwales/previousReleases> [Accessed 2 December 2020].
30. Oshiro, T. M., Perez, P. S., & Baranauskas, J. A. (2012, July). *How many trees in a random forest?*. In International workshop on machine learning and data mining in pattern recognition (pp. 154-168). Springer, Berlin, Heidelberg.
31. Policy 49: 12–18. 'Fuel Poverty Action Plan' (2018)
32. Review. London: Centre for Analysis of Social Exclusion, LSE.
33. Roberts, D., Vera-Toscano, E., & Phimister, E. (2015). *Fuel poverty in the UK: Is there a difference between rural and urban areas?*. Energy policy, 87, 216-223.
34. Roberts, D., Vera-Toscano, E., & Phimister, E. (2015). *Fuel poverty in the UK: Is there a difference between rural and urban areas?*. Energy policy, 87, 216-223.
35. Romero, J. C., Linares, P., & López, X. (2018). *The policy implications of energy poverty indicators*. Energy policy, 115, 98-108.
36. Rugg, J. J. (2020). *London boroughs' management of the private rented sector*.
37. Rugkasa, J., Shortt, N. K., & Boydell, L. (2007). *The right tool for the task: 'boundary spanners' in a partnership approach to tackle fuel poverty in rural Northern Ireland*. Health and Social Care in the Community, 15(3), 221e230
38. Tiefelsdorf, M., & Boots, B. (1997). *A note on the extremities of local Moran's I and their impact on global Moran's I*. Geographical Analysis, 29(3), 248-257. Bonnet, 2000
39. Walker, R., McKenzie, P., Liddell, C., & Morris, C. (2012). *Area-based targeting of fuel poverty in Northern Ireland: An evidenced-based approach*. Applied Geography, 34, 639-649.
40. Walker, R., McKenzie, P., Liddell, C., & Morris, C. (2014). *Estimating fuel poverty at household level: An integrated approach*. Energy and Buildings, 80, 469-479.

41. Wong, D.W. (2004). *The modifiable areal unit problem (MAUP)*. In *WorldMinds: geographical perspectives on 100 problems* (571-575). Springer: Dordrecht. Distribution of Efficiency Ratings across Boroughs (2017)