

Guidance

BohaoSu

2023-12-13

Contents

1	Initial project scope	2
1.1	Research Question:	2
1.2	Hypothesis:	2
1.3	Methodology:	3
1.4	Potential Limitation of data and methods	3
1.5	RMD environment configuration	4
2	Data Introduction	4
2.1	Downloading, Unzipping and loading the data	4
2.2	Data Description	5
2.3	NA values	5
2.4	Accuracy and Biasing	6
2.5	Coordinate Reference System (CRS)	6
3	Data Cleaning and Processing	6
3.1	Dealing with NAs in spatial and non-spatial dataset	6
3.2	Converting Datatype	7
3.3	Delete or Filter outliers	8
3.4	Data Format Normalization	8
3.5	Dealing with Repetitive or Unique rows	8
3.6	DATA Integration	9
4	Exploration Spatial Data Analysis (ESDA)	9
4.1	Distribution and coorelationship	9
4.2	Several Histograms	10
4.3	Spatial Distribution	11
4.4	Spatial Patterns	12
4.5	Spatial Autocorrelation	12
5	Variables Selection	12
5.1	Selecting Independent Variables based on ESDA and Research Question	12
5.2	DATA Normalization and Standardlization	12
5.3	Create Variables for Generalising several similar columns	13
6	Regression Modelling	14
6.1	Spatial Baseline Model	14
6.2	Training set and Testing set	14
6.3	Model Applying	14
7		14
7.1	14
7.2	15

7.3	Residuals Analysis	15
8	Conclusion	15
8.1	Summary	15
8.2	Research Limitation	15
8.3	Future Research	15

1 Initial project scope

```
library(broom)
library(car)
library(classInt)
library(corrplot)
library(crosstalk)
library(DiagrammeR)
library(dplyr)
library(fs)
library(geojsonio)
library(ggplot2)
library(ggmap)
library(here)
library(janitor)
library(maptools)
library(mapview)
library(OpenStreetMap)
library(patchwork)
library(plotly)
library(RColorBrewer)
library(readr)
library(rJava)
library(rgdal)
library(RSQLite)
library(rgeos)
library(sf)
library(sp)
library(spatstat)
library(spdep)
library(stringr)
library(tidyverse)
library(tmap)
library(tmaptools)
```

1.1 Research Question:

- What are the factors that might lead to xxxxxxxxxxxx scores in the xxxxxxxx city?

1.2 Hypothesis:

- Null hypothesis: There is complete spatial randomness. No statistical significance exists in a set of given observations. There is no pattern - i.e. complete spatial randomness - in our data. There is no relationship between exam scores and other observed variables across London.
- Alternative hypothesis: Our data does exhibit a pattern.

1.3 Methodology:

1. The first step is always cleaning and pre-processing data, which is the foundation for any kinds of analysis and modelling.
2. Exploration Data Analysis(histograms and Q-Q plots for statistical information, KDE for spatial distribution, DBSCAN for spatial clustering, etc.) need to be done both for non-spatial and spatial fields. This step would clarify the simple relationship and some features inside the data.
3. Based on research purpose, the regression model also needs two important prerequisite to guarantee its adaptability and rationality.
 - The first one is “The xxxxxx’s happening does have summarizable and discernible spatial distribution characteristics and spatial patterns.” This indicates whether a spatial analysis rather than purely quantitative analysis should be utilized to address the research question. Hence, Spatial patterns analysis like KDE or DBSCAN should be operated to check whether there is random occurrences for the xxxxxxxxxx or not. If the result is complete random distribution, I’ll just do the basic quantitative analysis based on the non-spatial data.
 - The second one is “Spatial location information does play as a crucial and indispensable variable when building regression models.” This means which regression model should be utilized to analysis and predict the xxxxxx. I suppose spatial autocorrelation methods should be used to examine the adaptability of Tobler’s Law.[[Tobler, 1970](#)] If there is no evidence showing geographical elements does affect the dependant variables distribution, then linear regression model or polynomial regression model should be the options. Otherwise, we should consider spatial information and select spatial regression models like spatial lag and spatial error models or geographically weighted regression models.
4. Afterwards, some advanced filtering or merging should be operated based on the ESDA, after which some cleansed columns and features could be extracted from the raw data and regarded as the independent variables for regression model to test the hypothesis. The variables selection process should also take some background context and research purpose into consideration.
5. Then, modelling part should be emphasized on which model should be selected. Regression Model selection will refer to all above previous analysis and prerequisites. After establishing a baseline model, the focus shifts to evaluating and refining this model. This involves comparing the baseline model’s performance against the spatial models, using metrics such as R-squared, AIC, or RMSE for validation and visualization. This process of model selection and refinement is central to achieving reliable and meaningful insights from the spatial analysis.
6. At the End, all results and features would be generalized and summarized, and a primary research conclusion will be drawn towards the initial question.

1.4 Potential Limitation of data and methods

1.4.1 Data Limitation

- Issues with Spatial Scale: The spatial scale of the data (e.g., geographic extent and resolution) can affect the analysis outcomes. Different spatial scales may reveal different patterns and relationships.
- Data Quality and Completeness: The dataset may have missing or inaccurate data, leading to biased analysis results. And we will also drop part of the data due to some NAs or administrative boundaries, which will also impact on dataset’s completeness.

1.4.2 Methods Limitation

- Non-independence of Spatial Data: Traditional non-spatial regression models often assume independence between observations, which may not hold true for spatial data. Spatial dependence between neighboring locations can impact the accuracy of the model.

- Spatial Autocorrelation in Residuals: In many regression models, spatial autocorrelation in residuals is considered a serious violation. If residuals from the model show spatial clustering, it might indicate that key variables are missing from the model.

1.5 RMD environment configuration

Before the specific illustration and analysis procedure, some environment configuration should be down to guarantee this .RMD file's robust on various platform and devices.

- Download .bib and .csl file remotely for reference

```
# download reference.bib remotely from my github
download.file("https://github.com/xxxxxxx.bib",
              destfile=here::here("reference.bib"))

# download reference.bib remotely from my github
download.file("https://raw.githubusercontent.com/BohaoSuCC/CASA0005BohaoSu/main/ucl-institute-of-education-harvard.csl",
              destfile=here::here("ucl-institute-of-education-harvard.csl"))
```

- Create Data Folder for Loading and Saving Data

```
# create the folder storing data for a better robust
folder_name <- "Data"

# get the root dir
root_dir <- here::here()

# construct the full path
folder_path <- file.path(root_dir, folder_name)

# check if the folder already exists
if (!dir.exists(folder_path)) {
  dir.create(folder_path)
  message("Folder '", folder_name, "' created at ", folder_path)
} else {
  message("Folder '", folder_name, "' already exists at ", folder_path)
}
```

2 Data Introduction

2.1 Downloading, Unzipping and loading the data

```
#Downloading the relating files and save and unzip it.
download.file("https://data.london.gov.uk/download/statistical-gis-boundary-files-london/9ba8c833-637",
              destfile=here::here("Data", "statistical-gis-boundaries-london.zip"))

listfiles<-dir_info(here::here("Data")) %>%
  dplyr::filter(str_detect(path, "london.zip")) %>%
  dplyr::select(path)%>%
  pull()%>%
  #print out the .gz file
  print()%>%
  as.character()%>%
  utils::unzip(exdir=here::here("Data"))

# reading the shp
```

```

Londonwards<-fs::dir_info(here::here("Data",
                                   "statistical-gis-boundaries-london",
                                   "ESRI"))%>%

  # $ means exact match
  dplyr::filter(str_detect(path,
                           "London_Ward_CityMerged.shp$"))%>%

  dplyr::select(path)%>%
  dplyr::pull()%>%
  # read in the file in
  sf::st_read()

# Reading the csv and Add na argument to make sure csv's robust
# replace all the nas as " "
data_test <- read_csv(here::here("Data","Evictions_20231212.csv"), na=c(" "))

LondonWardProfiles <- read_csv("https://data.london.gov.uk/download/ward-profiles-and-atlas/772d2d64-
                               col_names = TRUE,
                               locale = locale(encoding = 'UTF-8'))

# Reading the shp file
community_areas <- st_read(here::here("Data","geo_export_7fdf694c-62dd-4de4-8f17-0b5ca2408993.shp"))

```

2.2 Data Description

- The dataset is mainly about xxxxxxxxxxxx, containing xxxxxxxxxxxxxx in New York city. It is collected by xxxxxx and xxxx through xxxxx and published in the [xxxx's website](#).
- Another data is xxxxx.shp, containing geographical information features about xxxxxxxxx in xxxx city, which is published by xxxxx and can be public accessed through [xxxx's website](#).

2.3 NA values

In the dataset, the NA values could probably mean the missed data, unrecorded observations, inapplicable data points, etc.

```

# check all of the columns have been read in correctly
Column_type_list <- evictions_points %>%
  summarise_all(class) %>%
  pivot_longer(everything(),
               names_to="All_variables",
               values_to="Variable_class")
total_rows <- nrow(evictions_points)

# get the na values proportion of each column
Column_NA_ratio_list <- evictions_points %>%
  summarise_all(~sum(is.na(.))/total_rows) %>%
  pivot_longer(everything(),
               names_to="All_variables",
               values_to="Variable_NA_ratio")

# check the CRS and any error within spatial data
st_geometry(BoroughMap)

Column_type_list
Column_NA_count_list

```

From the statistical chart we could see there are totally xxxx rows(observations) containing NAs values. Technically, I don't think it is a high rate and these NA values could have a significant impact on my analysis.

Also, I am going to consider how to deal with those NA values with different solutions according to each column's role during my analysis. Anyway, the specific solutions to these NA values should align with research question and analysis requirement, so the detailed Data processing would be demonstrated in Data Cleaning and Processing.

2.4 Accuracy and Biasing

- Due to the absence of some accuracy information such as measurement errors, data validation processes, etc, I will focus on the biases of the data. According to the description on the website [xxxx's website](#), the purpose of collecting these data is mainly to xxxxxxxxxxxxxxxxxxxxxx, which might bring about the biases of not xxxxxxxxxxxxxxxxxxxxxx. However, I do not think this kind of biases will bring obvious and significant impact on analysis results and conclusions, even though the data collection methods do have limitation which I would elaborate detailedly afterwards.

2.5 Coordinate Reference System (CRS)

- Explain the coordinate reference system used in the data, including its type (such as geographic or projected coordinate system) and specific name (like WGS 84, UTM, etc.).

```
# transform the non-spatial data into spatial data based on columns 'Longitude' 'Latitude'
Airbnb <- read_csv("prac5_data/listings.csv") %>%
  st_as_sf(., coords = c("longitude", "latitude"),
           crs = 4326) %>%
  st_transform(., 27700) %>%
  # After, do some relevant filter for the useful info
  filter(room_type == 'Entire home/apt' & availability_365 == '365')

# Transform the CRS
sf_DATA_transformed <- st_transform(sf_DATA, crs = 32650)
```

- In this analysis, we have selected the [specify CRS, e.g., WGS 84, EPSG:4326] as our Coordinate Reference System (CRS). This CRS aligns well with our study's geographic scope that includes [mention the geographical extent, e.g., multiple countries, global analysis, etc.].
- Moreover, the impact of using [specify CRS] on my spatial analysis, especially in GWR where a spatial weight matrix really matters, is significant. And that requires distance measurement should be calculated, demonstrated and visualized precisely. Using projected CRS, I believe, should be a better choice for visualization, especially for some local-scale analysis and maps.

3 Data Cleaning and Processing

3.1 Dealing with NAs in spatial and non-spatial dataset

Some columns, such as xxxxxxxxxxxx, are extremely important that we couldn't extract any useful information if there are NA values. Besides, its high accuracy makes it harder to fill missing values, which leads us to nothing else but to drop them. Some of the columns, such like xxxx and some categorical data, we also could classify all the NA values as a new category. Some columns like xxxxxxxxxxxx, we could assume, based on the context of the study and common sense, that the missing values are 0. Although this approach may introduce some degree of inaccuracy, it is considered a practical solution since the proportion of NA values in these columns is very high. Therefore, dropping these columns outright would be an unwise decision.

```

# na.omit():      NA
#   object
#
library(dplyr)
DATA_cleaned <- na.omit(DATA)

# dplyr::filter():
#   .DATA      ...
#
DATA_cleaned <- DATA %>%
  dplyr::filter(!is.na(COLUMN_name))

# tidyr::replace_na():      NA
#   DATA      replace
#
DATA_cleaned <- DATA %>%
  replace_na(list(COLUMN = replacement_value))

```

```

# na.omit():      NA
#   object
#
library(dplyr)
DATA_cleaned <- na.omit(DATA)

# dplyr::filter():
#   .DATA      ...
#
DATA_cleaned <- DATA %>%
  dplyr::filter(!is.na(COLUMN_name))

# tidyr::replace_na():      NA
#   DATA      replace
#
DATA_cleaned <- DATA %>%
  replace_na(list(COLUMN = replacement_value))

```

3.2 Converting Datatype

And we will also convert some columns into specific datatype for more convenient processing.

```

# as.numeric():
#   x
#
DATA$COLUMN <- as.numeric(DATA$COLUMN)

#as.character():
#   x
#
DATA$COLUMN <- as.character(DATA$COLUMN)

#as.Date():
#   x      format
#   format = "%Y-%m-%d"
DATA$date_COLUMN <- as.Date(DATA$date_COLUMN, format = "%Y-%m-%d")

```

```
DATA <- DATA %>%
  mutate(COLUMN = str_replace_all(COLUMN, "\\$", "")) %>% # remove dollar sign
  mutate(COLUMN = str_replace_all(COLUMN, ",", "")) %>% # remove the comma
  mutate(COLUMN = as.numeric(COLUMN))
```

3.3 Delete or Filter outliers

```
#select all spatial feature with the city boundary and transform its CRS
BoroughMap <- LondonBoroughs %>%
  dplyr::filter(str_detect(GSS_CODE, "^E09"))%>%
  st_transform(., 27700)

#delete specific rows by filter and some arguments

DATA_cleaned <- DATA %>%
  dplyr::filter(COLUMN >= lower_limit, COLUMN <= upper_limit)

# only remain points which inside the boundary
BluePlaquesSub <- BluePlaques[BoroughMap, , op=st_within]
# to identify points completely within the borough outline, or a variety of other options such as st_

# plot the map to check to see that they've been removed
tmap_mode("plot")
tm_shape(BoroughMap) +
  tm_polygons(col = NA, alpha = 0.5) +
tm_shape(BluePlaquesSub) +
  tm_dots(col = "blue")
```

3.4 Data Format Normalization

```
#tolower():
# x
#
DATA$column <- tolower(DATA$column)

#toupper():
# x
#
DATA$column <- toupper(DATA$column)

#str_trim():      stringr
# string      side
# side = "both"
DATA$column <- str_trim(DATA$column)
```

3.5 Dealing with Repetitive or Unique rows

```
# make sure there is no more repetitive rows in the data
BluePlaques <- distinct(BluePlaques)

#duplicated():
# x
#
```



```
DATA_cleaned <- DATA[!duplicated(DATA), ]
```

```
#unique():
```

```
# x
```

```
#
```

```
DATA_cleaned <- unique(DATA)
```

3.6 DATA Integration

```
# non-spatial inner_join, right_join, full_join
```

```
DATA_combined <- left_join(DATA1, DATA2, by = c('SAME_COLUMN_NAME'='SAME_COLUMN_NAME'))
```

```
# spatial data join Argument could be
```

```
result <- st_join(x, y, op = st_intersects)
```

4 Exploration Spatial Data Analysis (ESDA)

Exploration Spatial Data Analysis (ESDA) plays an significant role in spatial regression modeling, primarily focusing on three key aspects. - First part contains acquiring statistical characteristics and distribution patterns of all non-spatial data, providing a solid foundation for selecting relevant independent variables for the model. This step is crucial for understanding the underlying structure and relationships within the data. - Second part involves cluster analysis to ascertain if the data exhibits spatial clustering, indicating non-random distribution across the space. This analysis verifies one of the prerequisites for spatial regression modeling, ensuring the data's suitability for such analysis. - Another essential prerequisite will be checked in part3, which is the impact of geographical spatial differences on certain dependent variables in spatial data. This is accomplished by conducting spatial autocorrelation analysis, which helps to confirm if spatial factors significantly influence the variables in question, thereby validating the use of spatial regression techniques.

4.1 Distribution and coorelationship

```
data_test$Latitude <- as.numeric(data_test$Latitude)
```

```
clean_data <- data_test %>%
```

```
  filter_all(all_vars(!is.na(.)))
```

```
ggplot(clean_data, aes(x = Latitude)) +
```

```
  geom_histogram(binwidth = 0.01, fill = "blue", color = "black", alpha=0.5)
```

```
# var1
```

```
ggplot(data_test, aes(x = Latitude)) +
```

```
  geom_histogram(binwidth = 1, fill = "blue", color = "black")
```

```
# var1 var2
```

```
ggplot(data, aes(x = var1, y = var2)) +
```

```
  geom_point()
```

```
# - data num_var
```

```
ggplot(data, aes(y = num_var)) +
```

```
  geom_boxplot(fill = "lightblue", color = "blue")
```

```
# Bar charts
```

```
ggplot(data, aes(x = cat_var)) +
```

```

    geom_bar(fill = "lightgreen", color = "darkgreen")

# cormatrix---- data
numeric_data <- data[c("num_var1", "num_var2", "num_var3")]
#
cor_matrix <- cor(numeric_data)
#
corrplot(cor_matrix, method = "circle")
# numeric_data
# corrplot()
#method = "circle"
#"circle"
#"square" "circle"
#"ellipse"
#"number"
#"shade"
#"color"
#"pie"

```

4.2 Several Histograms

```

names(data_test)

## [1] "Court Index Number"      "Docket Number"
## [3] "Eviction Address"       "Eviction Apartment Number"
## [5] "Executed Date"          "Marshal First Name"
## [7] "Marshal Last Name"      "Residential/Commercial"
## [9] "BOROUGH"                "Eviction Postcode"
## [11] "Ejectment"              "Eviction/Legal Possession"
## [13] "Latitude"               "Longitude"
## [15] "Community Board"        "Council District"
## [17] "Census Tract"           "BIN"
## [19] "BBL"                    "NTA"

columns_name_datatest <- names(data_test)[c(13,14,15,16)]

# create a new list to save every histogram
plots_list <- list()

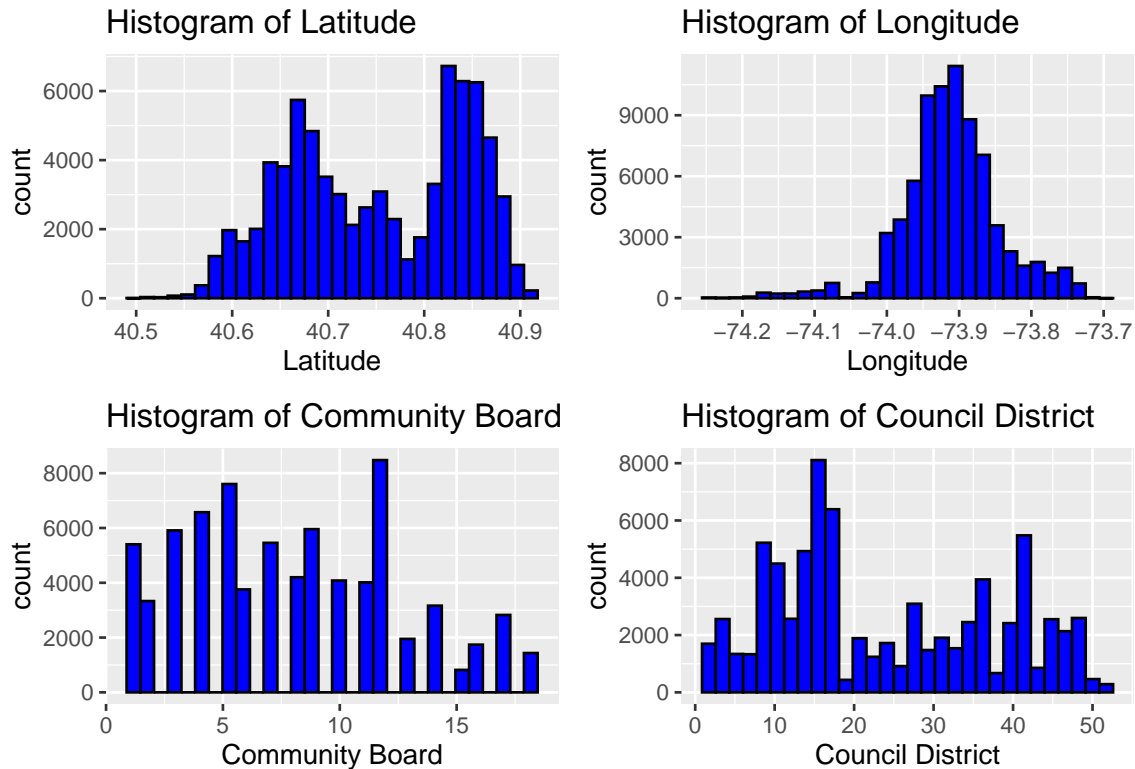
#
#ggplot(clean_data, aes(x = Latitude)) +
#  geom_histogram(binwidth = 0.05, fill = "blue", color = "black", alpha=0.5)

# create a histogram towards every column and add it into the plot list
for (i in 1:4) {
  column_name <- columns_name_datatest[i]
  p <- ggplot(data_test, aes(x = !!sym(column_name))) +
    geom_histogram(fill = "blue", color = "black") +
    ggtitle(paste("Histogram of", column_name)) #
  plots_list[[i]] <- p
}

# using patchwork to combine every histograms into a 2*2 grids
combined_plot <- wrap_plots(plots_list, ncol = 2)

```

combined_plot



4.3 Spatial Distribution

In order to understand the spatial distribution, I utilized Kernel Density Estimation (KDE) for its analysis. This method could provide with an in-depth examination of _____ (specific aspect or feature of the data), enabling a clearer visualization of spatial concentration and patterns within the study area. The KDE plot is particularly effective in highlighting areas of high density or clustering, which is essential for the analysis of _____ (specific phenomena or geographic feature).

```
ggplot(DATAFRAME, aes(x = longitude, y = latitude)) +
  geom_point() +
  theme_minimal() +
  labs(x = "Longitude", y = "Latitude", title = "Spatial Scatter Plot")
```

```
ggplot(df, aes(x = longitude, y = latitude)) +
  stat_density_2d(aes(fill = ..level..), geom = "polygon") +
  scale_fill_viridis_c() +
  labs(x = "Longitude", y = "Latitude", title = "Spatial Heatmap") +
  theme_minimal()
```

- From the Heatmaps plot we could see xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

```
ggplot(df, aes(x = longitude, y = latitude)) +
  stat_density_2d(aes(fill = ..level..), geom = "polygon") +
  scale_fill_viridis_c() +
  labs(x = "Longitude", y = "Latitude", title = "Spatial Heatmap") +
  theme_minimal()
```

- From the KDE (Kernel Density Estimation) plot, we can observe _____. This sug-

gests that the distribution of data exhibits a trend of _____, which is crucial for understanding _____ (specific data characteristic or geographic feature). Particularly in the region of _____, this distribution provides key insights into _____ (a relevant phenomenon or issue).”

4.4 Spatial Patterns

```
#create a sp object
BluePlaquesSub<- BluePlaquesSub %>%
  as(., 'Spatial')

#create a ppp object
BluePlaquesSub.ppp <- ppp(x=BluePlaquesSub@coords[,1],
                          y=BluePlaquesSub@coords[,2],
                          window=window)

#first extract the points from the spatial points data frame
BluePlaquesSubPoints <- BluePlaquesSub %>%
  coordinates(.)%>%
  as.data.frame()

#now run the DBSCAN analysis
db <- BluePlaquesSubPoints %>%
  fpc::dbscan(.,eps = 700, MinPts = 4)

#now plot the results
plot(db, BluePlaquesSubPoints, main = "DBSCAN Output", frame = F)
plot(BoroughMap$geometry, add=T)

BluePlaquesSubPoints%>%
  dbscan::kNNdistplot(.,k=4)
```

4.5 Spatial Autocorrelation

5 Variables Selection

5.1 Selecting Independent Variables based on ESDA and Research Question

LASSO

5.2 DATA Normalization and Standardlization

```
#
normalize <- function(x) {
  return ((x - min(x)) / (max(x) - min(x)))
}

DATA_normalized <- as.DATA.frame(lapply(DATA, normalize))

# 2. Standardization
# 0 1
DATA_standardized <- scale(DATA)
#scale(x, center = TRUE, scale = TRUE)
```

```

#x
#center = TRUE
#scale = TRUE

#caret
# preProcess
library(caret)

preProcValues <- preProcess(DATA, method = c("range"))
preProcValues <- preProcess(DATA, method = c("center", "scale"))
DATA_normalized <- predict(preProcValues, DATA)
#preProcess(x, method)
#x
#method = c("range")
#method = c("center", "scale")

library(dplyr)
library(purrr)

DATA_normalized <- DATA %>%
  mutate_if(is.numeric, normalize)

DATA_standardized <- DATA %>%
  mutate_if(is.numeric, scale)

```

5.3 Create Variables for Generalising several similar columns

```

# 1.
# R      +, -, *, /
DATA$new_var = DATA$var1 + DATA$var2

# 2.  dplyr mutate
#dplyr mutate
library(dplyr)
DATA <- DATA %>%
  mutate(new_var = var1 / var2)

# 3.
# ifelse dplyr case_when
DATA$new_var = ifelse(DATA$var1 > threshold, value_if_true, value_if_false)

DATA <- DATA %>%
  mutate(new_var = case_when(
    condition1 ~ value1,
    condition2 ~ value2,
    TRUE ~ default_value
  ))

# 4.
# lubridate

library(lubridate)
DATA$year <- year(DATA$date_var)
DATA$month <- month(DATA$date_var)

```

```

# 5.
# stringr
library(stringr)
DATA$new_var = str_sub(DATA$text_var, 1, 5) #

# 6.
# factor dplyr mutate as.factor
DATA$new_var = as.factor(DATA$var1)
DATA <- DATA %>%
  mutate(new_var = as.factor(var1))

# 7.
# scale
DATA$new_var = scale(DATA$var1, center = TRUE, scale = TRUE)

# 8.
# dplyr group_by summarize
DATA_summary <- DATA %>%
  group_by(group_var) %>%
  summarize(mean_var = mean(var1, na.rm = TRUE))

# 9.
#R          log, exp, mean, median
DATA$log_var = log(DATA$var1)

```

6 Regression Modelling

6.1 Spatial Baseline Model

Establishing a spatial baseline model typically refers to creating a basic regression model that accounts for spatial variability. This model serves as a benchmark for comparison, allowing the evaluation of the GWR model or other spatial models against non-spatial models, like ordinary least squares regression. The spatial baseline model usually includes variables relevant to the study but does not incorporate treatments for spatial variability, thus providing a clear view of the model's performance changes after introducing the spatial dimension.

```
#
```

6.2 Training set and Testing set

```
#
```

6.3 Model Applying

7

7.1

```
# K          MSE    R2
```

7.2

```
# Grid Search Random Search
```

7.3 Residuals Analysis

In spatial regression models, ensuring that residuals are normally distributed is important because many statistical inferences (like tests for the significance of parameters) are based on the assumption of normality. If residuals are not normally distributed, this can affect the reliability of the model and the validity of the conclusions. Therefore, using a Q-Q plot to examine the normality of residuals in spatial regression models is a crucial step in model diagnostics.

```
# A Q-Q plot is a common method to check if data follows a normal distribution. In a Q-Q plot of a s
# Identifying Outliers: The Q-Q plot can also help identify outliers in the data. Data points that de
```

```
# Create the dataframe for normal distribution reference
data <- rnorm(100) # using the normal distribution random generated numbers.
df <- data.frame(sample = data)

ggplot(df, aes(sample = sample)) +
  stat_qq() +
  stat_qq_line(col = "red") +
  ggtitle("Q-Q Plot") +
  xlab("Theoretical Quantiles") +
  ylab("Sample Quantiles")
```

7.3.1

```
#
```

8 Conclusion

8.1 Summary

8.2 Research Limitation

8.3 Future Research

- Geography Detector
-

References

W. R. Tobler. A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography*, June 1970. URL <https://www.tandfonline.com/doi/abs/10.2307/143141>. Publisher: Routledge.