MTHM506/COMM511 - Statistical Data Modelling

2024-03-22

Tuberculosis (TB) poses a significant challenge in Brazil due to its high prevalence and far-reaching impact. As one of the countries with the highest TB rates globally, Brazil grapples with a complex web of health, social, and economic consequences. Beyond its toll on individual health, TB strains Brazil's healthcare system, leading to increased healthcare costs and resource allocation challenges. Moreover, TB exacerbates social inequalities by disproportionately affecting vulnerable populations and perpetuating the cycle of poverty. The disease is often accompanied by stigma and discrimination, hindering efforts to seek diagnosis and treatment. Addressing TB is not only a matter of public health but also crucial for social cohesion and economic development. Furthermore, in an interconnected world, TB poses risks to global health security, necessitating collaborative efforts to control its spread. Brazil's commitment to TB research and innovation plays a vital role in advancing diagnostics, treatments, and preventive measures, contributing to global efforts to eliminate TB and improve public health outcomes worldwide.

Objectives

Analyse the spatial and temporal influences of socioeconomic indicators on the rate of TB prevalence. We will utilise the GAM framework which will allow us to build upon the linear model framework. We will evaluate the significance of the variables and how they are correlated with the rate of TB prevalence. We would like to determine whether there have been variables excluded from our dataset in predicting rates of TB and suggest some potential reasons for any variance not explained by our model.

The dataset for TB cases in Brazil is comprised of 1,671 observations across 14 variables. For us to begin looking at the impact of socio-economic factors on the prevalence of TB within different regions, we had to create a new variable called 'rate'. This rate is defined as the number of cases of TB per unit 10,000 people living in the region. We will then go on to model how these variables are correlated with the rate of TB prevalence and assess which of the factors are most significant in determining changes in TB prevalence.

The GAM model can be expressed mathematically as:

$$\log(\mu_i) = f_1(\text{Illiteracy}_i, \text{Urbanisation}_i, \text{Poor\_Sanitation}_i, \text{Poverty}_i) + f_2(\text{Urbanisation}_i) + \alpha_{\text{RegionCluster}_i} + f_3(\text{Timeliness}_i) + f_4(\text{Density}_i)$$

where: - μ_i is the expected number of TB cases (rate) for the i-th observation. - $f_1(\cdot)$ represents a tensor product smooth function for the interaction between Illiteracy, Urbanisation, Poor Sanitation, and Poverty. It allows for the modeling of complex, non-linear interactions between these variables. - $f_2(\text{Urbanisation}_i)$ is a cubic regression spline for Urbanisation, capturing its non-linear effect on the TB rate. - $\alpha_{\text{RegionCluster}_i}$ is a fixed effect for each level of the categorical variable RegionCluster, indicating that the model includes separate intercepts for different clusters of regions. - $f_3(\text{Timeliness}_i)$ and $f_4(\text{Density}_i)$ are smooth functions (potentially using splines) for Timeliness and Density, respectively, to capture their non-linear relationships with the TB rate. - The response variable ($\log(\mu_i)$) is modeled on the log scale due to the use of a negative binomial regression framework, which is appropriate for count data and accounts for overdispersion. The link function here is logarithmic, relating the linear predictor to the mean of the negative binomial distribution.

Parameters and Functions

- The $f(\cdot)$ terms represent smooth functions, implemented using splines in GAMs. These functions are flexible and can model complex relationships without specifying a predefined form.
- k indicates the basis dimension for the splines, controlling the smoothness of the fitted function. Higher k values allow for more complex shapes.
- bs = "cr" specifies that cubic regression splines are used for the smooth terms.
- The model is estimated using Restricted Maximum Likelihood (REML) with a selection criterion to determine the significance of predictors, ensuring a balance between model fit and complexity.

This mathematical description outlines how the model captures both linear and non-linear effects of predictors on TB rates, interactions among predictors, and variations across different regions, all within a framework suitable for count data with potential overdispersion.

For us to determine which of the socio-economic factors most significantly influence the rate of TB prevalence, we will need to construct a model. We will construct a General Additive Model (GAM). A GAM model expands on the linear model framework, however, allows for extra flexibility when modelling non-linear relationships between the dependent and independent variables. The framework of these models will allow us to analyse and discuss spatial and temporal patterns in our data, which would otherwise not be captured by more simplistic linear model framework. The interaction between variables in epidemiological studies often tends to exhibit nonlinear patterns, which GAM models can help to manoeuvre around, enabling us to accurately analyse these nonlinear trends. The relationship between each of the covariates is depicted in the correlation matrix (above/below).

```
##                 summary_data$coefficients[, "Pr(>|t|)"] coef(model)
## (Intercept)                                1.684215e-05 -1.65506708
## Indigenous                                 1.018848e-03  0.03572553
## Illiteracy                                 7.524614e-02 -0.01567351
## Urbanisation                               1.176369e-04  0.01659446
## Density                                    2.238923e-10  2.02534999
## Poverty                                    9.437384e-02  0.01045893
## Poor_Sanitation                            1.564034e-05 -0.02461527
## Unemployment                               7.014726e-14  0.15107369
## Timeliness                                 9.877524e-13  0.01369027
```

Each predictor's coefficient represents the estimated change in the TB rate (per 10,000 people) for each unit increase in the predictor, holding all other predictors constant.

Significant predictors ($p < 0.05$) include:

Indigenous: Positively associated with TB rate.

Urbanisation: Positively associated with TB rate.

Density: Strong positive association with TB rate.

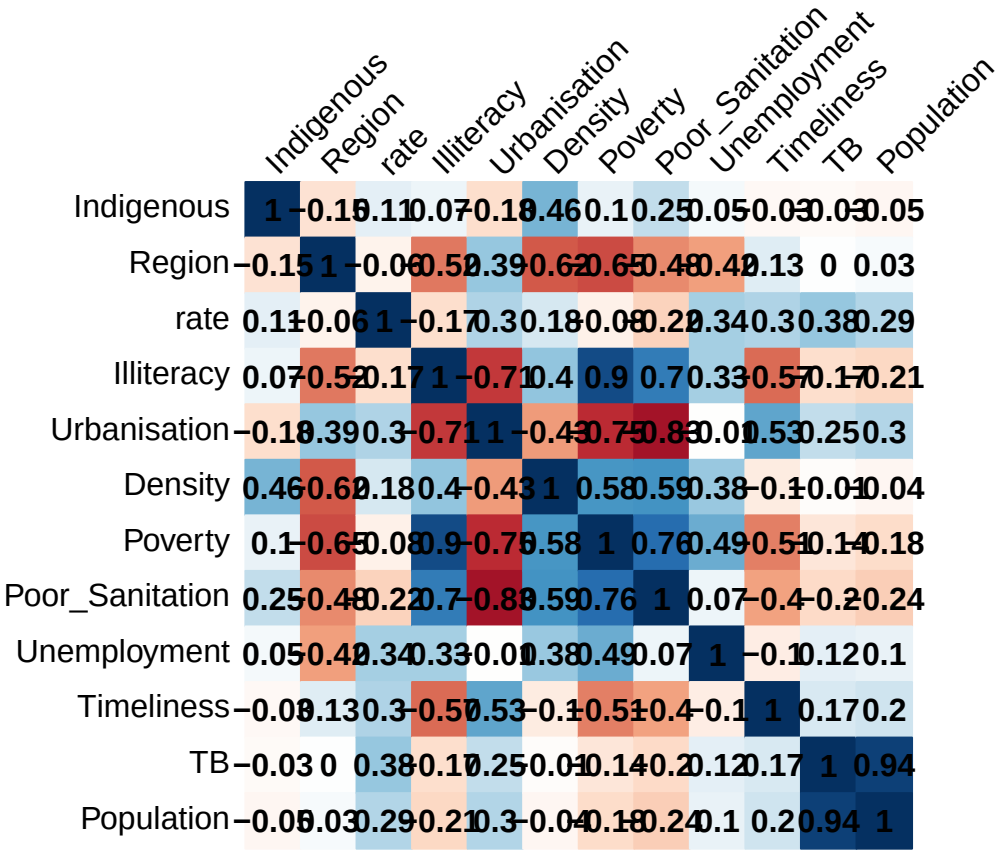
Poor Sanitation: Negative association with TB rate.

Unemployment: Positive association with TB rate.

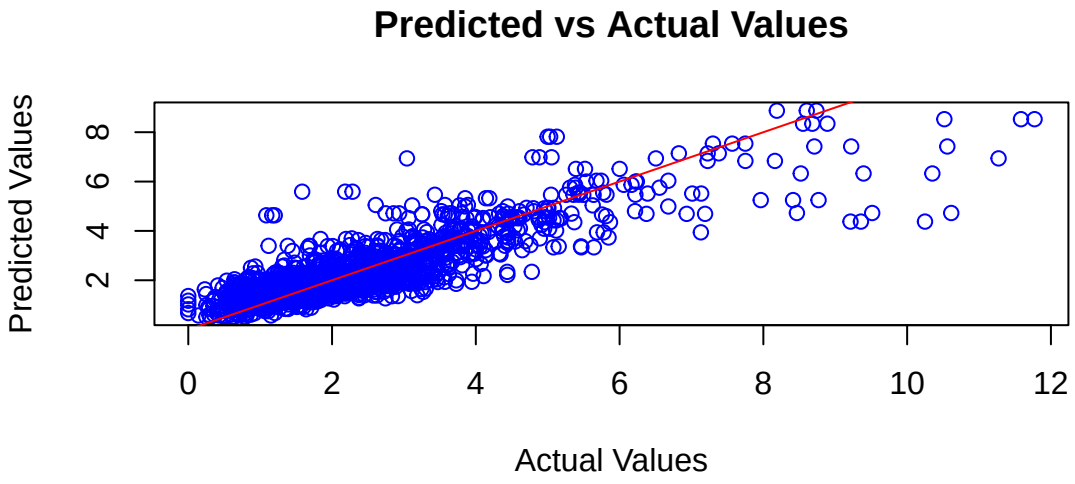
Timeliness: Positive association with TB rate.

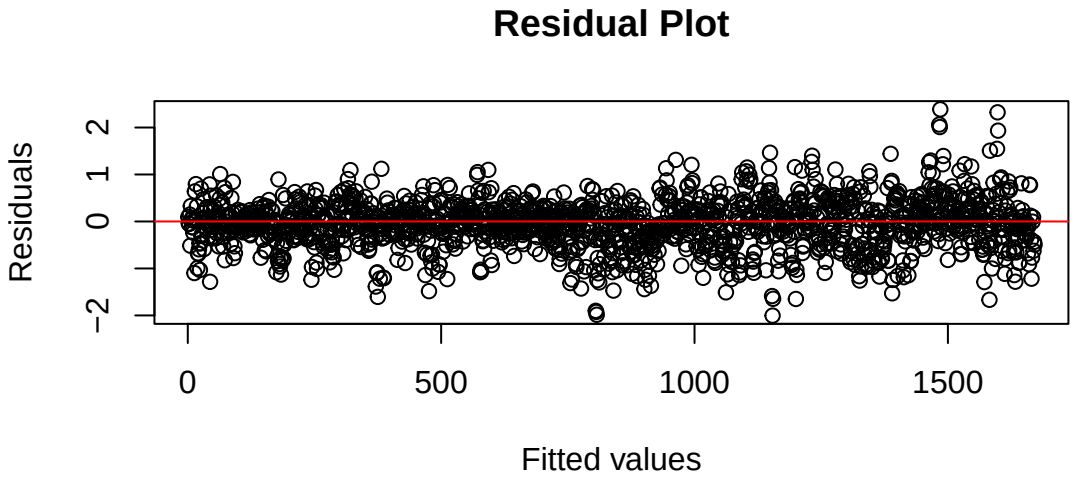
The model explains about 29.44% of the variability in TB rates (Multiple R-squared), which is good but suggests other factors not included in the model also influence TB rates. To figure out other factors would require deeper analysis. An example of an indicator which may have been neglected in our dataset is HIV prevalence. HIV makes individuals susceptible to contracting TB, and areas with higher HIV rates often have higher TB rates.

The F-statistic's p-value is less than 2.2e-16, indicating that the model is statistically significant overall.

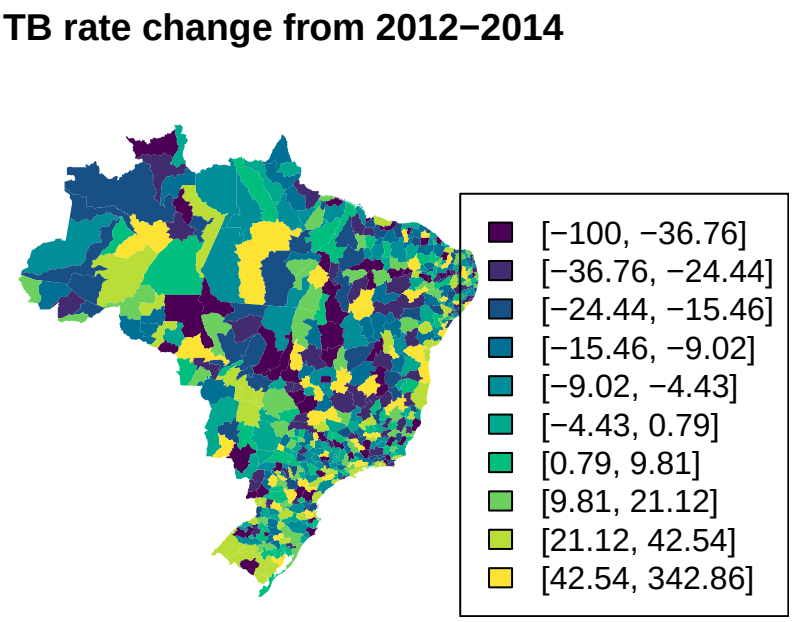


Before conducting any thorough analysis, we were thinking that urbanisation will be slightly positively correlated with the rate of TB, with some lag factor involved. Since TB is an airborne disease, we would expect that as regions become more densely populated, we would expect the disease to be more transmissible. We expect each of the following to be more significantly related to the rate of tb prevalence, but to all be positively correlated: illiteracy, density, poverty, unemployment, timeliness, and poor sanitation. These all seem like sensible predictions, for example, education and poverty levels are commonly positively associated with the spread of disease. We expect timeliness to be negatively associated with rates of TB, since it is used as an indicator of the responsiveness of the healthcare system.

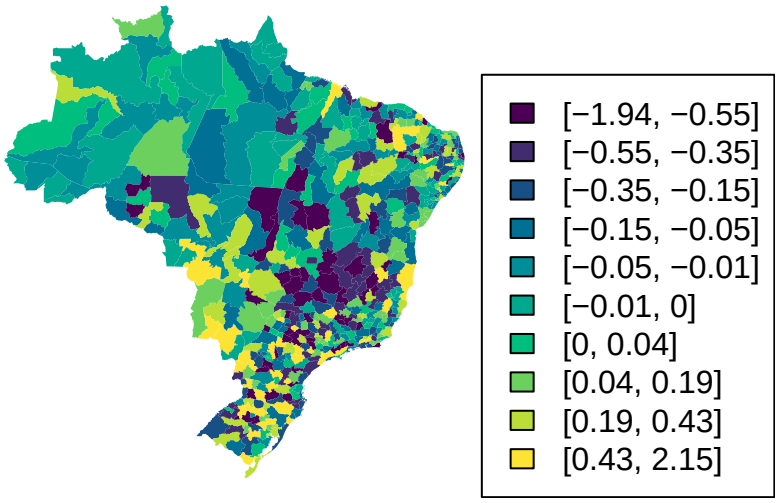




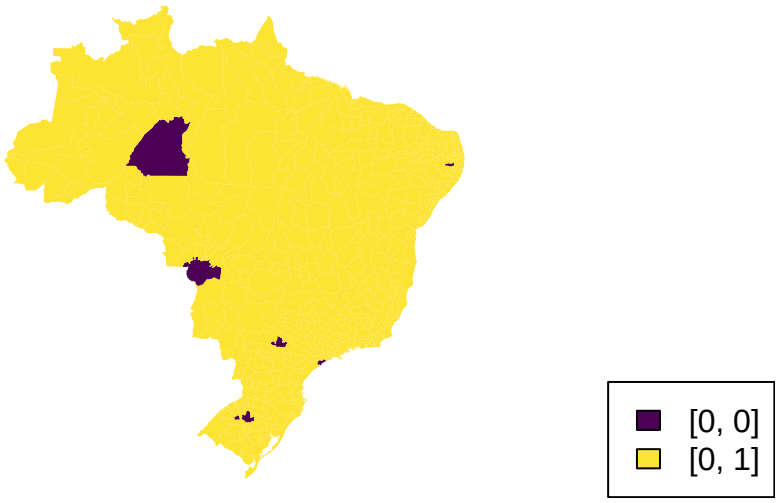
The model demonstrates high accuracy in predicting tuberculosis (TB) risk at lower levels, as indicated by the close alignment of data points to the red line of perfect prediction on the scatter plot. However, the model's predictive accuracy diminishes at higher levels of TB risk. This trend is visible in the plot, where the proximity of points to the red line decreases as the actual TB risk values increase. Consequently, the divergence of points from this line at elevated risk values suggests that the model may need refinement to improve its predictive performance in these higher-risk regions.



The spatio-temporal analysis reveals intriguing patterns in the change of TB rates over time. Notably, the southern regions exhibit a concerning trend of increasing rates, contrasting with the marked improvements observed in the northern regions. Upon further examination of the mean TB rates across regions over time, a nuanced narrative emerges. Despite the substantial decrease in TB rates observed in the northern regions, they continue to grapple with relatively higher rates compared to other parts of the country. Conversely, while the southern regions experience higher increases in TB rates, their overall rates remain comparatively low. This multifaceted insight underscores the complexity of TB dynamics across different regions, necessitating targeted interventions tailored to the unique challenges and trends observed within each geographical context.



By taking the mean of the residuals over the years and visualizing it geospatially, we were able to identify regions with higher unexplained risk, which could provide valuable insights for further investigation or targeted interventions. It's evident that regions in the south and east coast exhibit a larger range in residuals, indicating higher unexplained variance compared to the north of the country. This information can guide public health officials and policymakers in focusing resources and efforts on these high-risk regions to mitigate the spread of TB and improve healthcare capabilities. We have identified the regions which have seen the most drastic increases in the risk of TB throughout the populations. We notice that the southern regions are suffering from TB outbreaks, and there also appears to be a cluster in the northwest of Brazil. These regions have also seen some of the sharpest increases in the risk of TB. For these reasons, it would be sensible for public health authorities to pay close attention to these regions. It's possible that these regions have smaller healthcare infrastructure in terms of hospitals and doctors. Regions with relatively fewer establishments could be considered more demanding of additional funds. We could also bear in mind that areas with high values of timeliness may be operating closer to their peak capability, and that there could be many unreported and undiagnosed cases in regions where the timeliness values are lower. Overall, we need to be sure to consider the marginal benefits of any additional fundings to any region. We should compare how their current healthcare infrastructure may be improved while also considering that the TB rates are unlikely to be fully reported in areas of low timeliness.




```
##         Attribute Means Region.13007 Region.26016 Region.35035 Region.35056
## 1    Urbanisation 71.96        93.82        87.14        85.19        95.22
## 2    Unemployment  6.93        10.48        16.58         5.86        10.02
## 3      Illiteracy 14.80         4.91        15.99         7.81         6.12
## 4 Poor_Sanitation 16.45         4.89         7.95         4.79         1.39
## 5         Poverty 44.37        41.11        58.67        21.08        33.89
## 6      Timeliness 47.67        75.54        62.04        93.38        72.33
##   Region.43025 Region.51017
## 1        79.78        95.35
## 2         6.45         6.54
## 3         7.33         5.41
## 4         7.95         3.47
## 5        29.61        24.55
## 6        70.58        52.70
```

To assist the health authorities in Brazil in determining which regions required assistance in resources to combat their high TB rates, we first had to calculate the top 1% of TB Rates in Brazil in 2014. As can be seen in the plot above, there are 6 regions spread throughout Brazil that qualify in that 1%. They have exceedingly high rates of Tuberculosis among their populations and require urgent assistance from the authorities. We tested the different variables involved in our dataset on each of the 6 regions to determine which region required the most urgent assistance. Region 26016 is clearly the region that requires the most immediate assistance, with the poverty rate, poor sanitation, illiteracy, and unemployment rates all being higher than the other 5 regions.

```
##
```

```
##  Moran I test under randomisation
##
## data:  model_residuals
## weights: weights
##
## Moran I statistic standard deviate = 17.513, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.2751297649     -0.0005988024      0.0002478839
```

```
##
##  Geary C test under randomisation
##
## data:  model_residuals
## weights: weights
##
## Geary C statistic standard deviate = 13.247, p-value < 2.2e-16
## alternative hypothesis: Expectation greater than statistic
## sample estimates:
## Geary C statistic       Expectation          Variance
##      0.7190610599      1.0000000000      0.0004497696
```

The Moran's I and Geary's C tests provided significant insights into the spatial correlation of tuberculosis (TB) risk across regions, which played a crucial role in our decision to incorporate region as a fundamental component of our model. Specifically, the Moran's I test yielded a statistic of 0.275 with a p-value less than 2.2e-16, strongly indicating spatial autocorrelation and suggesting that TB rates are not randomly distributed across regions but instead exhibit a significant pattern of clustering. Similarly, the Geary's C test, with a statistic of 0.719 and the same p-value threshold, reinforced this finding by indicating a deviation from spatial randomness. These statistical tests underscore the importance of considering spatial relationships and regional effects when modeling TB rates, thereby justifying the inclusion of region as a key factor in our analytical framework.

Overall, we need to be sure to consider the marginal benefits of any additional fundings to any region. We should compare how their current healthcare infrastructure may be improved while also considering that the TB rates are unlikely to be fully reported in areas of low timeliness.

We can conclude that our GAM model is rigorous due to it producing a high adjusted R squared value and concluding that each of the variables were significant in predicting rates of TB. We can see from the residual plot that at lower risk of TB prevalence, we are better able to predict future rates of TB. However, we can see that at higher rates, the model loses accuracy in predictions. This is a clear indicator that there is some noise or information that is not being captured in explaining the rate of TB, which helps to answer one of our objectives.

Bibliography

```
# Load necessary libraries for data manipulation and visualization
library(fields)
library(maps)
library(sp)
library(GGally)
library(ggplot2)
```

```r
library(corrplot)
library(dplyr)
library(tidyr)
library(lmtest)
library(mgcv)
library(spdep)  # Load the 'spdep' package
library(car)

# Set a seed for reproducibility of results that involve random processes
set.seed(19032024)

# Load dataset from specified path
load("C:/Users/archi/Downloads/datasets_project.RData")

# Creating a new variable 'rate' in TBdata dataframe, calculated as the incidence of TB per 10,000 indi
TBdata$rate <- (TBdata$TB/TBdata$Population)*10000

# Generate summary statistics and inspect the first few rows of the dataset to understand its structure
summary(TBdata)
head(TBdata)

# Print the names of the columns in the TBdata dataframe
print(colnames(TBdata))

# Checking for missing values across all columns in TBdata, summarizing the count of NA values per colu
missing_values <- sapply(TBdata, function(x) sum(is.na(x)))

# Checking for negative values in specific columns that logically should only contain non-negative valu
columns_to_check <- c('Population', 'TB', 'Poverty', 'Poor_Sanitation', 'Unemployment')
negative_values <- sapply(TBdata[columns_to_check], function(x) sum(x < 0))

# Print the counts of missing and negative values
print(missing_values)
print(negative_values)

# Compute a correlation matrix for selected variables in the dataset to explore relationships
cor_matrix <- cor(TBdata[, c("Indigenous",'Region',"rate", "Illiteracy", "Urbanisation", "Density", "Po

# Display the correlation matrix using colored cells to represent the magnitude of correlations; annota
corrplot(cor_matrix, method="color", addCoef.col = "black", tl.col="black", tl.srt=45, cl.pos="n")

# Zero out the diagonal and upper triangle of the correlation matrix to avoid redundancy and self-corre
cor_matrix[lower.tri(cor_matrix, diag = TRUE)] <- 0

# Identify and extract pairs of variables with high correlation (absolute value greater than 0.7)
high_cor_indices <- which(abs(cor_matrix) > 0.7, arr.ind = TRUE)
high_cor_variables <- data.frame(
  row = rownames(cor_matrix)[high_cor_indices[, 1]],
  col = colnames(cor_matrix)[high_cor_indices[, 2]]
)

# Filter out self-correlations from the pairs of variables with high correlations
high_cor_variables <- subset(high_cor_variables, row != col)
```

```r
# Print pairs of variables that have a high correlation
print(high_cor_variables)

# Build a linear regression model with 'rate' as the dependent variable and several predictors
model <- lm(rate ~ Indigenous + Illiteracy + Urbanisation + Density + Poverty + Poor_Sanitation + Unempl

# Display a summary of the model, including coefficients and their statistical significance
summary(model)

# Generate an ANOVA table for the linear model to assess the overall significance of the model
anova(model)

# Perform a Type II ANOVA which is preferred for unbalanced designs or when the model includes interact
Anova(model, type="II")

# Checking assumptions of linear regression by visualizing residuals to assess homoscedasticity (equal
par(mfrow=c(2,2))  # Setup the plotting area to display 4 plots in a 2x2 grid
plot(model)  # Generate diagnostic plots for the regression model

# Extract coefficients from the model
coefficients <- coef(model)
print(coefficients)

# Further summarization of the model, including extracting p-values for the significance of each coeffi
summary_data <- summary(model)
p_values <- as.data.frame(summary_data$coefficients[, "Pr(>|t|)"])

# Extract coefficients from the model object
coefficients_data <- as.data.frame(coef(model))

# Combine the p-values and coefficients into a single data frame for easier comparison and interpretati
combined_data <- cbind(p_values, coefficients_data)
print(combined_data)

# Use ggpairs to create a matrix of scatterplots for pairs of variables in the dataset, useful for visu
ggpairs(TBdata)

# Define a subset of variables of interest for further analysis
variables_of_interest <- c("Indigenous", "Illiteracy", "Urbanisation", "Density", "Poverty", "Poor_Sani

# Recompute the correlation matrix for selected variables and display it using corrplot for visual anal
cor_matrix <- cor(TBdata[, c("Indigenous",'Region',"rate", "Illiteracy", "Urbanisation", "Density", "Po
corrplot(cor_matrix, method="color", addCoef.col = "black", tl.col="black", tl.srt=45, cl.pos="n")


# Assuming 'TBdata' has latitude and longitude for each observation, combine these into coordinates
coordinates <- cbind(TBdata$lon, TBdata$lat)

# Create a neighbors list using k-nearest neighbors method. Here, each location is considered to have 4
# This neighbors list defines which observations are considered spatially related.
nb <- knn2nb(knearneigh(coordinates, k = 4))

# Convert the neighbors list into a weights matrix, using row-standardization ("W"). This step assigns
```

```r
weights <- nb2listw(nb, style = "W")

# Assuming 'gam_model' is a previously fitted spatial regression model, extract the residuals
model_residuals <- residuals(gam_model)

# Perform Moran's I test on the model's residuals using the defined weights. Moran's I is a measure of
# Positive values indicate a tendency of similar values to cluster spatially, while negative values sug
moran.test(model_residuals, weights)

# Perform Geary's C test on the model's residuals, also using the same weights. Geary's C is another me
# but it is more sensitive to local variations. Values significantly different from 1 indicate spatial
geary.test(model_residuals, weights)

# Aggregate TBdata by 'Region' and compute the mean of several variables, ignoring NA values
region_data <- TBdata %>%
  group_by(Region) %>%
  summarise(
    Indigenous = mean(Indigenous, na.rm = TRUE),
    Illiteracy = mean(Illiteracy, na.rm = TRUE),
    Urbanisation = mean(Urbanisation, na.rm = TRUE),
    Density = mean(Density, na.rm = TRUE),
    Poverty = mean(Poverty, na.rm = TRUE),
    Poor_Sanitation = mean(Poor_Sanitation, na.rm = TRUE),
    Unemployment = mean(Unemployment, na.rm = TRUE),
    Timeliness = mean(Timeliness, na.rm = TRUE),
    Population = mean(Population, na.rm = TRUE)
  )

# Standardize the selected variables to have a mean of 0 and a standard deviation of 1
ndata <- scale(select(region_data, Indigenous, Illiteracy, Urbanisation, Density, Poverty, Poor_Sanitat

# Perform hierarchical clustering on the scaled dataset
hc <- hclust(dist(ndata))  # Compute a distance matrix and perform hierarchical clustering

# Plot the resulting dendrogram to visualize the clustering and help decide on the number of clusters
plot(hc)

# Cut the dendrogram to form a specified number of clusters, here illustrated as k = 300 (likely a typo
groups <- cutree(hc, k = 300)

# Add the region and cluster information back into the aggregated data for merging
region_data$Region <- as.character(region_data$Region)  # Ensure 'Region' is treated as a character var
region_data$RegionCluster <- groups  # Add cluster assignments to region_data

# Merge the cluster assignments back into the original TBdata based on 'Region'
TBdata <- merge(TBdata, select(region_data, Region, RegionCluster), by = "Region", all.x = TRUE)

# Fit a generalized additive model (GAM) to the data, specifying a complex model structure including no
gam_model <- gam(rate ~ te(Illiteracy, Urbanisation, Poor_Sanitation, Poverty, k = 4) +  # Tensor produ
                   s(Urbanisation, k = 10, bs = "cr") +  # Cubic regression spline for Urbanisation
                   as.factor(RegionCluster) +  # Treat RegionCluster as a categorical factor
                   s(Timeliness, k=4) +  # Smooth for Timeliness
                   s(Density, k=4),  # Smooth for Density
```

```
                data = TBdata, method = "REML", select = TRUE,  # REML for model fitting, 'select' for
                family = nb(link = 'log'))  # Negative binomial family for count data

# Display a summary of the fitted GAM, showing effects of predictors, smooth terms, and their significa
summary(gam_model)

#######
# Vis #
#######

# Predict TB rates using the fitted GAM model
predicted <- predict(gam_model, type = "response")

# Plot actual TB rates against the predicted rates
plot(TBdata$rate, predicted,
     xlab = "Actual Values", ylab = "Predicted Values",
     main = "Predicted vs Actual Values",
     col = "blue", pch = 1)
abline(a = 0, b = 1, col = "red")  # Adds a red line y = x for reference


# Plot residuals to assess how well the model fits the data
plot(residuals(gam_model), ylab = "Residuals", xlab = "Fitted values", main = "Residual Plot")
abline(h = 0, col = "red")  # Adds a horizontal red line at y = 0

# Store the model's residuals in the TBdata dataframe for further analysis
residuals_data <- residuals(gam_model)
TBdata$resid <- residuals_data

# Visualize the spatial distribution of model residuals for the year 2014
plot.map(TBdata$resid[TBdata$Year==2014], n.levels=7, main="Resid counts for 2014")

# Repeat the visualization for the year 2013
plot.map(TBdata$resid[TBdata$Year==2013], n.levels=7, main="Resid counts for 2013")

# Repeat the visualization for the year 2012
plot.map(TBdata$resid[TBdata$Year==2012], n.levels=7, main="Resid counts for 2012")


# Store the predicted TB rates from the model into the TBdata dataframe
TBdata$Predicted_Rate <- predicted
# Visualize the spatial distribution of predicted TB rates for the year 2014
plot.map(TBdata$Predicted_Rate[TBdata$Year==2014], n.levels=7, main="Pred rate counts for 2014")

# Repeat the visualization for the year 2013
plot.map(TBdata$Predicted_Rate[TBdata$Year==2013], n.levels=7, main="Pred rate counts for 2013")

# Repeat the visualization for the year 2012
plot.map(TBdata$Predicted_Rate[TBdata$Year==2012], n.levels=7, main="Pred rate counts for 2012")


# Assuming your dataset is named 'data_set'
# Make sure the data is ordered appropriately
```

```r
TBdata <- TBdata %>%
  arrange(Region, Year)

# Calculate the percentage change in rate
TBdata <- TBdata %>%
  group_by(Region) %>%
  mutate(percentage_change = (rate - lag(rate)) / lag(rate) * 100) %>%
  ungroup()

plot.map(TBdata$percentage_change[TBdata$Year==2014],n.levels=7,main="percentage_change counts for 2014
plot.map(TBdata$percentage_change[TBdata$Year==2013],n.levels=7,main="percentage_change counts for 2013


# Filter for years 2012 and 2014, then reshape the data so each year's rate is in its own column using
data_2012_2014 <- TBdata %>%
  # Filter rows to only include data from the years 2012 and 2014
  filter(Year %in% c(2012, 2014)) %>%
  # Reshape data: create columns for each year with TB rates for those years
  pivot_wider(names_from = Year, values_from = rate)

# Calculate the percentage change from 2012 to 2014 for each region
percentage_change_2012_2014 <- data_2012_2014 %>%
  # Calculate percentage change in TB rates from 2012 to 2014 and add it as a new column
  mutate(percentage_change = (`2014` - `2012`) / `2012` * 100)

# Select only the relevant columns, if necessary
percentage_change_2012_2014 <- percentage_change_2012_2014 %>%
  # Keep only the 'Region' and the calculated 'percentage_change' columns
  select(Region, percentage_change)
data_2012_2014 <- data_2012_2014 %>%
  # Filter out rows where percentage change could not be calculated (i.e., any NAs)
  filter(!is.na(percentage_change))

# Visualize the percentage change in TB rates from 2012 to 2014
plot.map(data_2012_2014$percentage_change, n.levels=10, main="TB change from 2012-2014")


TBdata$Year <- as.factor(TBdata$Year)

# Summarize TB data by calculating mean and standard deviation of rates by Region and Year
mean_std_data <- TBdata %>%
  # Group data by both 'Region' and 'Year' to calculate statistics within these groups
  group_by(Region, Year) %>%
  # Calculate mean and standard deviation of TB rates, ignoring NA values
  summarise(mean_rate = mean(rate, na.rm = TRUE),
            sd_rate = sd(rate, na.rm = TRUE),
            # Drop grouping structure automatically once summarization is done
            .groups = "drop") %>%
  # Explicitly ungroup data to avoid accidental carry-over of grouping
  ungroup()

# Calculate the percentage change in mean TB rate across years for each region
percentage_change_data <- mean_std_data %>%
```

```r
  # Group by 'Region' to calculate changes within each region across years
  group_by(Region) %>%
  # Calculate percentage change in mean TB rate from the previous year for each region
  # 'lag' function is used to get the previous year's mean_rate
  # 'default = first(mean_rate)' ensures the calculation starts correctly from the first available year
  mutate(percentage_change = (mean_rate / lag(mean_rate, default = first(mean_rate))) * 100 - 100) %>%
  # Remove grouping after calculation
  ungroup()

# Prepare a summary table organizing calculated values for easier comparison
summary_table <- percentage_change_data %>%
  # Select only relevant columns for the summary table
  select(Region, Year, mean_rate, sd_rate, percentage_change) %>%
  # Reshape data to have one row per region and separate columns for each year's statistics
  pivot_wider(names_from = Year, values_from = c(mean_rate, sd_rate, percentage_change), names_sep = "_
  # Rename columns to clarify whether they represent rates or other statistics
  # Adds ' (per year)' to rate-related columns for clarity
  rename_all(~ paste0(.x, ifelse(grepl("rate", .x), " (per year)", "")))

# Display the summary table
print(summary_table)

# Replace infinite values with NA in the summary table
summary_table_no_inf <- summary_table %>%
  mutate_all(~replace(., is.infinite(.), NA))

# Remove rows containing any NA values to clean the data further
summary_table_no_inf <- summary_table_no_inf %>%
  drop_na()

# Alternative approach combining replacement of infinite values with NA and removal of any NAs in one s
summary_table_clean <- summary_table %>%
  mutate_all(~replace(., is.infinite(.), NA)) %>%
  na.omit()

# Calculate the mean of all columns in the summary table, ignoring NAs
summary_means <- summary_table %>%
  summarise_all(mean, na.rm = TRUE)

# Identify columns related to mean TB rates across different years
mean_rate_columns <- grep("mean_rate_", colnames(summary_table), value = TRUE)

# Calculate the average of these mean rates for each region, adding it as a new column
summary_table$mean_mean_rate <- rowMeans(summary_table[mean_rate_columns], na.rm = TRUE)

# Visualize the average TB rate across the specified years for each region
plot.map(summary_table$mean_mean_rate, n.levels=10, main="TB rate mean over 2012-2014")

# Calculate the mean residual from the model for each region
mean_residuals <- TBdata %>%
  group_by(Region) %>%
  summarise(mean_residual = mean(resid, na.rm = TRUE)) %>%
  ungroup()
```

```r
# Display the calculated mean residuals for each region
mean_residuals

# Visualize the mean residuals for each region, showing the model's error distribution spatially
plot.map(mean_residuals$mean_residual, n.levels=10, main="model residuals over 2012-2014")

# Calculate the 99th percentile of TB rates for 2014 to identify the threshold for the top 1%
top_1_percent_threshold <- quantile(TBdata$rate[TBdata$Year == 2014], 0.99)

# Create a binary variable in the dataset indicating whether a region's TB rate falls within the top 1%
TBdata$Top_1_Percent <- with(TBdata, ifelse(Year == 2014 & rate >= top_1_percent_threshold, 1, 0))

# Calculate the total count of regions that fall within the top 1% TB rates for 2014
top_1_percent_count <- sum(TBdata$Top_1_Percent)

# Extract various attributes for regions in the top 1%, such as Timeliness, Poverty, etc.
top_1_percent_timeliness <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Timeliness")]
top_1_percent_poverty <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Poverty")]
top_1_percent_poorsanitation <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Poor_Sanitation")]
top_1_percent_unemployment <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Unemployment")]
top_1_percent_illiteracy <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Illiteracy")]
top_1_percent_urban <- TBdata[TBdata$Top_1_Percent == 1, c("Region", "Urbanisation")]

# Variables to display the extracted information for the regions in the top 1%
top_1_percent_timeliness
top_1_percent_poverty
top_1_percent_poorsanitation
top_1_percent_illiteracy
top_1_percent_unemployment
top_1_percent_urban

# Calculate mean values for various attributes for all regions in 2014
mean(TBdata$Timeliness[TBdata$Year==2014])
mean(TBdata$Poverty[TBdata$Year==2014])
mean(TBdata$Poor_Sanitation[TBdata$Year==2014])
mean(TBdata$Urbanisation[TBdata$Year==2014])
mean(TBdata$Unemployment[TBdata$Year==2014])
mean(TBdata$Illiteracy[TBdata$Year==2014])

# Plot the regions with top 1% TB rates for 2014, assuming 'plot.map' visualizes spatial data
plot.map(TBdata$Top_1_Percent[TBdata$Year == 2014], n.levels = 2, main = "Top 1% TB Rate for 2014")


# Define the attributes and their mean values for 2014
attributes <- c("Urbanisation", "Unemployment", "Illiteracy", "Poor_Sanitation", "Poverty", "Timeliness

mean_values_2014 <- c(71.96, 6.93, 14.80, 16.45, 44.37, 47.67)



# Define the values for each region
values_13007 <- c(93.82, 10.48, 4.91, 4.89, 41.11, 75.54)
values_26016 <- c(87.14, 16.58, 15.99, 7.95, 58.67, 62.04)
```

```r
values_35035 <- c(85.19, 5.86, 7.81, 4.79, 21.08, 93.38)
values_35056 <- c(95.22, 10.02, 6.12, 1.39, 33.89, 72.33)
values_43025 <- c(79.78, 6.45, 7.33, 7.95, 29.61, 70.58)
values_51017 <- c(95.35, 6.54, 5.41, 3.47, 24.55, 52.70)



# Create a data frame for the table

# Set up a data frame to compare mean attribute values for 2014 against specific regions' values
df_table <- data.frame(
  Attribute = attributes,
  Means = mean_values_2014,
  `Region-13007` = values_13007,
  `Region-26016` = values_26016,
  `Region-35035` = values_35035,
  `Region-35056` = values_35056,
  `Region-43025` = values_43025,
  `Region-51017` = values_51017
)

# Print the comparative data frame
print(df_table)
```