CA2: Individual Project

SN: 740096324

Problem and Data Exploration

This project specifically aims to quantify tuberculosis (TB) risk, and as a result we must initially standardise the response variable `TB` by region population size; our new response variable is TB per 100,000 (see C.2). When beginning to tackle a problem of this scale, it is important to understand the shape of our data and the distributions of our socio-economic covariates, of which we have 8: Indigenous, Illiteracy, Urbanisation, Density, Poverty, Poor Sanitation, Unemployment and Timeliness. Note that since the values of our covariates for each region are constant across years (asserted in C.3), only values for `Year == 2014` are used throughout socio-economic covariate exploration and model building. *Table 1* details some key summary statistics for each of these covariates, as well as TB per 100,000. *Figure 1* then presents frequency histograms for each of these variables. Only Indigenous and to a lesser extent Density and TB per 100,000 present distributions largely deviating from the Normal, all with significant positive skew. This is confirmed by high values for skew ($> |1|$) in *Table 1*. Whilst smoothing in general additive models is generally robust to highly varied covariate distributions, extreme skew may still result in splines that poorly fit the sparser data region.

Another key factor to understand in our exploration of this dataset is the presence of correlations between our predictive covariates. *Figure 2* presents a correlation matrix visualising the absolute values of the Pearson correlation coefficient r for each pairwise combination of covariates. By taking the absolute value, we can see specifically those combinations with the strongest linear relationships, regardless of direction. 5 pairs of covariates return values for r much above the rest, and are restricted to pairs between just 4 of the 8 covariates (Poverty ~ Illiteracy = 0.897; Poor Sanitation ~ Urbanisation = 0.834; Poor Sanitation ~ Poverty = 0.761; Poverty ~ Urbanisation = 0.750; Illiteracy ~ Urbanisation = 0.707). As a result, it is highly likely that there is a degree of interdependency between these 4 covariates, which we must be mindful of when designing our final model.

Lastly, *Figure 3* visualises the spatial distribution of TB cases per 100,000 population. By adjusting the colour scale to fixed boundaries rather than evenly-spaced quantiles, we are better able to identify the microregions with the most extreme risk. It is clear that microregions with the highest risk fall into one of two categories: the larger areas of sparser population in the north-west, and compact microregions spread across the east coast and the southern tip of Brazil - likely to be urban metropolises. It is important to note this structural variation in region size and thus population density, as we will later be modelling the impacts of latitude and longitude on TB risk. A theoretical frequency histograms of Latitude and Longitude would indicate skew towards the east and south respectively; microregions are not Normally distributed geographically. Furthermore, as we can see in *Table 2*, there also exist some correlations between the Longitude and Latitude values for the centroids of each microregion and our 8 socio-economic covariates. Notable are strong positive relationships between the Latitude and Poverty ($r = 0.779$), Illiteracy ($r = 0.690$) and Density ($r = 0.644$). We know from before that Poverty and Illiteracy are strongly correlated, suggesting a spurious correlation here between Illiteracy and Latitude, and we can see in *Figure 3* evidence that Density varies substantially across Brazil. All of these correlations are considerations that can be returned to when devising an suitable model.

Model Fitting

The initial step in fitting a generalised additive model (GAM) is to select the most appropriate family for capturing the distribution of the response variable, which in our case is TB per 100,000. Our response

variable is continuous, immediately ruling out those suitable for count data, such as the Poisson and Negative Binomial families, as well as binary data, such as the Binomial family. Furthermore, we saw earlier in *Figure 1* that the distribution of `TB_per_100000` is positively skewed, suggesting the Gaussian family will fit poorly. As such, the 2 families under serious consideration are the Gamma and Tweedie families. Both models are inherently suited towards overdispersed data, yet they contrast in a few key manners: the Gamma family assumes continuous data where the Tweedie accepts a structural combination of continuous and count data; the Gamma family requires strictly positive response variable where the Tweedie family is suited towards excess zero-values; and the Gamma family assumes $\sigma^2 \propto \mu^2$ where the Tweedie family more flexibly assumes $\sigma^2 \propto \mu^p; 1 < p \leq 2$. From these listed differences, it does initially appear the Gamma family may be slightly more appropriate, however our response variable contains some zero-values (see *Table 1*). This is not allowed when modelling off the Gamma distribution, but can be alleviated through an ϵ-adjustment of 0.0001 to each zero-value (see C.2), thus allowing us to compare these two models.

All family model comparison is performed upon data for only 2014 with a basic covariate set-up: all 8 socio-economic variables included with a smoothing term applied to each (see C.9). As is clear from the comparative QQ plots in *Figure 4*, the vast majority of the data is well-modelled by the Tweedie family, unlike the Gamma family. There is clear evidence of heavy positive and negative tails, but compared to the Gamma family model, the underlying distribution is clearly captured better. The residuals vs. fitted plots of *Figure 4* tell are less contrasting, with both models presenting an even spread of residual variance across the range of fitted values, save for a single outlier in the Gamma model. Studying the summary outputs of both models shows a clear preference towards the Tweedie family also. Both the adjusted R^2 and deviance explained values exceed that over the Gamma family model ($0.356 > 0.344$; $42.3\% > 34.8\%$); this suggests that the Tweedie family captures the variance in the observed data better, important as we are prioritising a model with high predictive power. The Gamma family does return a much lower Generalised Cross-Validation (GCV) score ($0.30039 < 1.0972$), suggesting it may outperform the Tweedie model at predicting on unseen data, yet I consider this insufficient at outweighing the listed benefits of using the Tweedie family. For the above reasons, I selected the Tweedie family as the underlying distribution of my final model.

The first step in tweaking this initial model is to optimise our estimate of p - the power parameter. I repeated this initial model across a range of values (1.1 - 1.9), and produced various summary model comparison statistics highlighted in *table 3*. I selected $p = 1.3$ as this maximised the explained deviance %; this is the metric that best captures the model's predictive power, particularly for datasets such as ours where the response variable is significantly non-Normal.

From this new baseline, I then tweaked the model in several subsequent ways, elucidated in C.13. Initially, I removed all univariate smooths of Illiteracy, Poverty and Urbanisation, and replaced these with a single tensor smooth of their interactions. This is because our previous model showed non-significant p-values for these covariates' F-statistics, and we saw before in *Figure 2* that these covariates are highly correlated. I then modified the Indigenous covariate via a Box-Cox transformation to minimise the effect of the extreme skew we noted in *Figure 1*. Next, I hypothesised that there may be an interaction effect between Density and Poor Sanitation, as contextually, poor sanitation may arise in both high density urban areas as well as remote, low density rural areas. Therefore, I added a tensor interaction smooth for these covariates. I then noted via `k.check()` that both Indigenous and Timeliness exhibited significant p-values for presence of too few basis dimensions – thus I implemented `k=50` for each smooth. Lastly, I hypothesised a potential interaction between Indigenous and Unemployment, with the assumption that indigenous people face different structural causes of unemployment. Again, I added a tensor interaction smooth to incorporate this. At each of the above steps, I ensured that the implemented change both increased the explained deviance and decreased AIC, so that I could be confident that predictive power was increasing alongside goodness-of-fit.

With this detailed model on the socio-economic covariates in hand, I could then evaluate the contribution of spatial, temporal and spatio-temporal effects to unexplained risk. To do so I added 3 more smoothing terms: a thin-plate regression spline smooth across Latitude and Longitude, a cubic regression spline for Year (limited to 3 basis dimensions), and tensor smooth for these 3 covariates with their aforementioned spline choices. Applying the `summary()` function to this model indicated that the Year term was insignificant, and therefore I removed this term from the final model.

Final Model

Our final model relies upon the Tweedie distribution with a log link function, selected for its strength in handling a mixed discrete-continuous response variable that exhibits evidence of overdispersion, striking a balance between the Poisson and Gamma distributions. It is defined by its variance-mean relationship (see below), which can be flexibly edited via the power parameter p to suit the observed data distribution from count-like at $p \approx 1$ to Gamma-like at $p = 2$.

$$\text{Var}(Y) = \phi\mu^p, \quad 1 < p \leq 2$$

The final model is listed below:

```
final_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous_BoxCox, k=50) + s(Density) +
                           s(Poor_Sanitation) + s(Unemployment) + s(Timeliness, k=50) +
                           te(Illiteracy, Poverty, Urbanisation) +
                           ti(Density, Poor_Sanitation) +
                           ti(Indigenous_BoxCox, Unemployment) +
                           s(lat, lon, bs = "tp") +
                           te(lat, lon, Year, bs = c("tp", "tp", "cr"), k = c(10, 10, 3)),
                           data = TBdata,
                           family = Tweedie(p = 1.3, link = power(0)))
```

Each included covariate and interaction returned a p-value < 0.05 for the F-statistic, justifying their inclusion in the model through their significant effect on variance explanation. The key model comparison summary statistics are: adjusted-r^2: 0.78, explained deviance: 78.7%, and GCV: 1.083. The final QQ and residuals vs. fitted plots are displayed in *Figure 5*. These indicate a slightly poorer fit from our initial model, with heavier tails on the QQ plot, yet the variation is residuals is still consistent across fitted values. Nonetheless, the final model approximates our data sufficiently, and explains a high proportion of the observed deviance.

Results and Conclusions

From my finalised model, I can conclude to the Brazilian health authorities that each of the 8 socio-economic variables significantly contribute to the risk of tuberculosis in varying manners. Rates of Indigenous, Density, Poor Sanitation, Unemployment and Timeliness each contribute individually to explaining variance in risk. There is also an interaction effect between the remaining 3 covariates: Illiteracy, Poverty and Urbanisation, each of which are highly correlated with each other. Lastly, there are two further interaction effects present, between: Density and Poor Sanitation, and Indigenous rates and Unemployment. These effects total to explaining 51.4% of deviance ($r^2 = 0.445$, GCV $= 1.970$). Furthermore, I conclude that there both spatial and spatio-temporal effects explain further variation in TB risk, yet there is no evidence that temporal effects, when isolated, contribute to explaining the observed trends. When incorporating spatial and spatio-temporal effects into our model, we can explain 78.7% of deviance ($r^2 = 0.78$, GCV $= 1.083$). Therefore, it can be stated that our model accounts for a substantial proportion of observed variance, modelling the dataset effectively.

As a result, we can make predictions on future TB risk. By calculating predicted values on transposed data for 2015 (see C.16), we can assess which microregions we predict will present the greatest risk of TB spread, and thus should be priority targets for the health authorities. These results are visualised in *figure 6*. From these predictions, the three microregions with the greatest risk are: `13007`, `51017` and `35063`, each with a predicted risk of over 90 cases per 100,000 population. The full list of the top 10 microregions sorted by risk are asserted in C.17. It would be possible to extrapolate our findings further into the future to assess more prolonged temporal risk, however, you must consider that there are only 3 values for Year in our observed data. This limits the model's predictive power across time, and thus it would be advisable for the health authorities to continue data collection for further years in order to provide more thorough evidence for how risk changes with time, particularly with respect to location.

Critical Review

I see several aspects of the modelling process as being successful. I believe I made an accurate choice from the exponentional dispersion family, basing this choice upon structural features of the response variable, contextual clues of the overall problem, and goodness-of-fit metrics. The Tweedie family modelled our data with sufficient accuracy, and handled the significant overdispersion of our response variable well. Many of these choices were also backed up with appropriately-chosen and well-designed visualisations that highlighted the key context behind the data. In particular the frequency histograms and correlation matrix shed meaningful insight into the distributions of our covariates. I also believe I made good use of model summary statistics, interpreting their outputs accurately and helping to tweak my model towards its optimal form. This includes maintaining focus on which statistics were proportionally more important to the defined problem objectives, allowing me to better tweak the model for my specific goals.

On the other hand, I believe my conclusions could have been served better by a more thorough understanding of the technicalities behind the various regression splines and smooth processes. It is likely that the application of more precise choices for these two aspects of generalised additive models would have supported the final model in modelling more optimally, and thus producing more accurate risk predictions. This consideration goes hand-in-hand with the idea that I likely would have benefitted from a deeper understanding of what to look for when deciding which covariates require interaction terms to be accounted for. I relied upon contextual clues alongside observed correlations, yet I recognise this process could have been formalised better. Lastly, there is the possibility of further improving the final model through a process that would utilise all 3 years of available data. Whilst temporal aspects were studied, they played no part in the model tuning process, meaning there was significant quantity of data that was not taken advantage of at that stage. This could have been done via some form of cross-validation or setting an offset for Year, but I was not able to piece together a statistical justification for a method that does so.

References

Wood, S. N. (2017). *Generalized Additive Models: An Introduction with R* (2nd ed.). CRC Press.

Appendix A - Figures

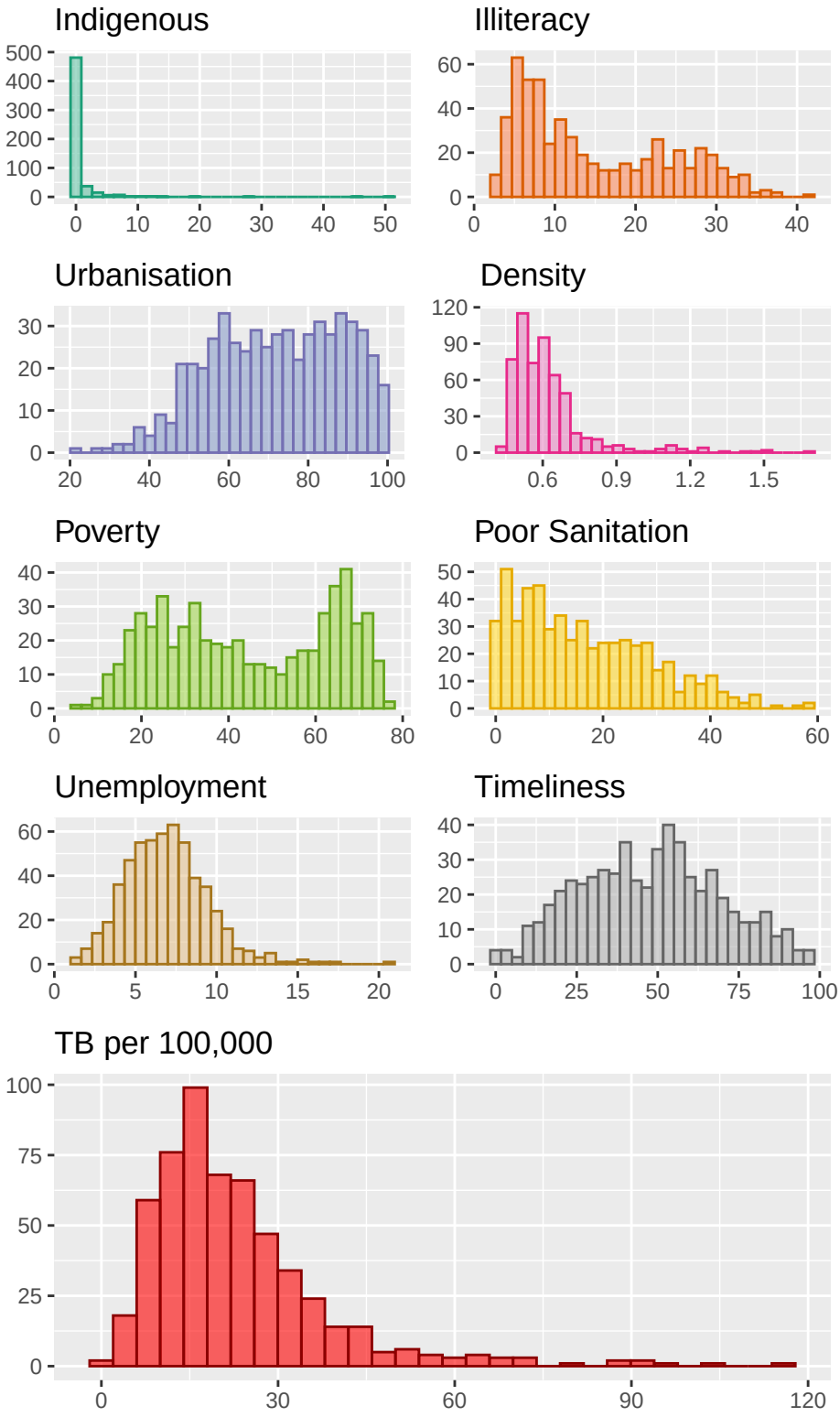

Figure 1: Frequency histograms for the 8 socio-economic covariates, as well as the response variable TB per 100,000.

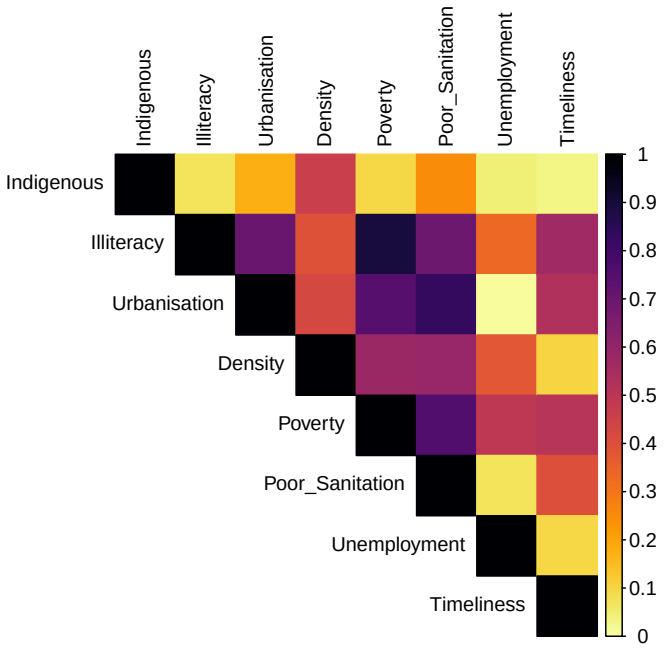

Figure 2: Correlation Matrix highlighting the pairwise R^2 values between each of the 8 socio-economic covariates.

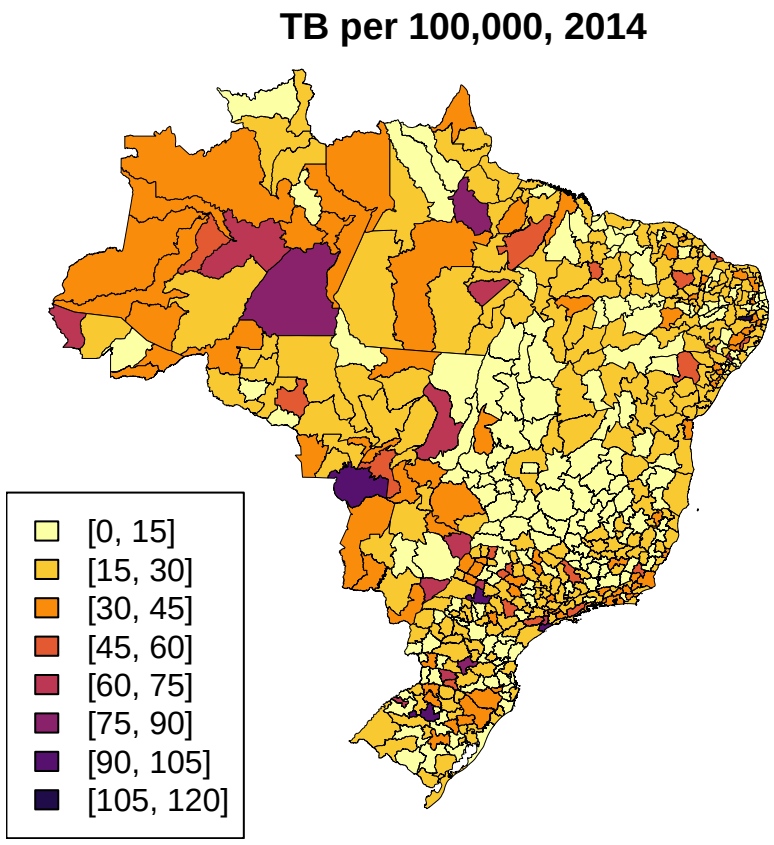

Figure 3: TB risk (counts per 100,000 population) across the 557 microregions of Brazil.

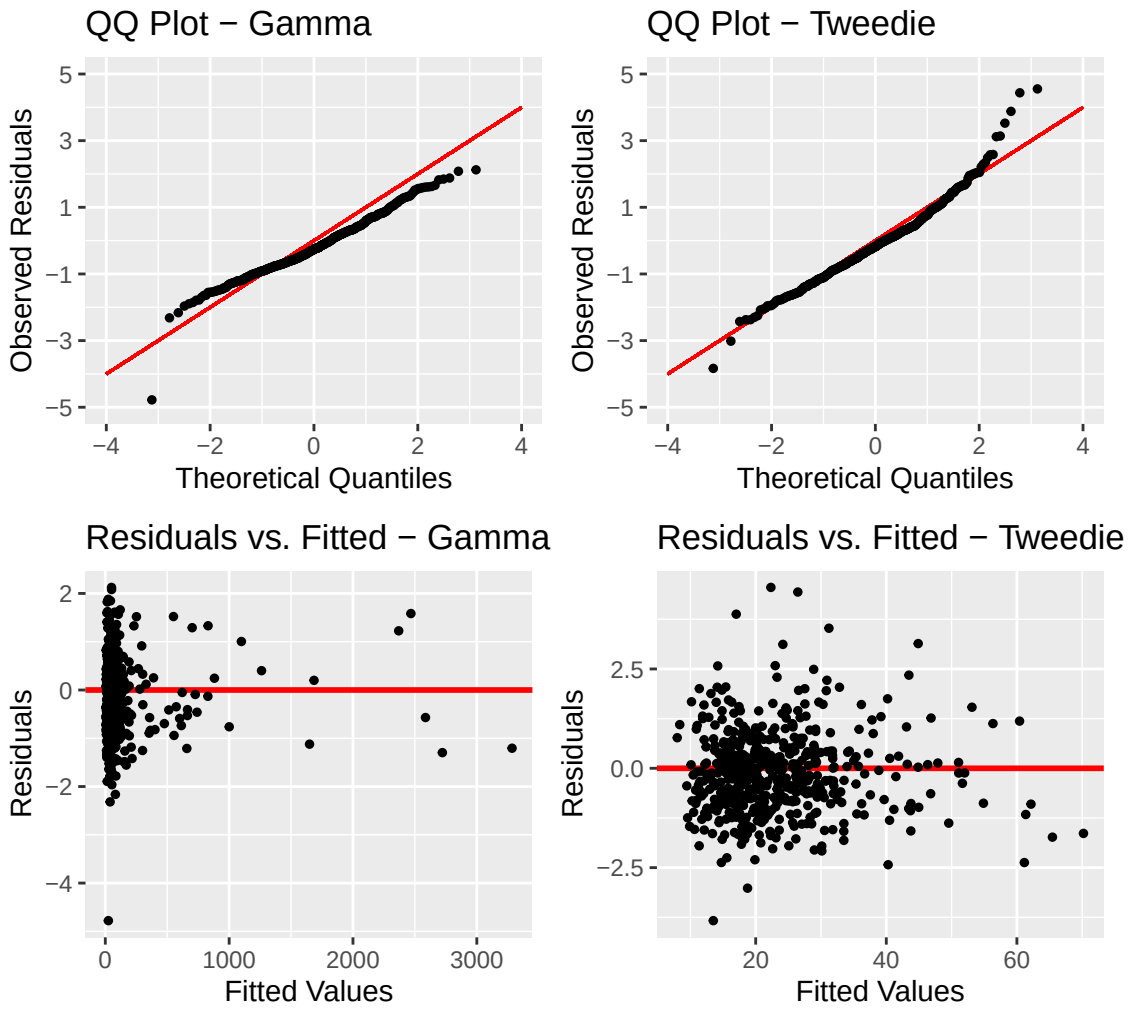

Figure 4: QQ plots and Residuals vs. Fitted plots for each of the initial Gamma and Tweedie family models.

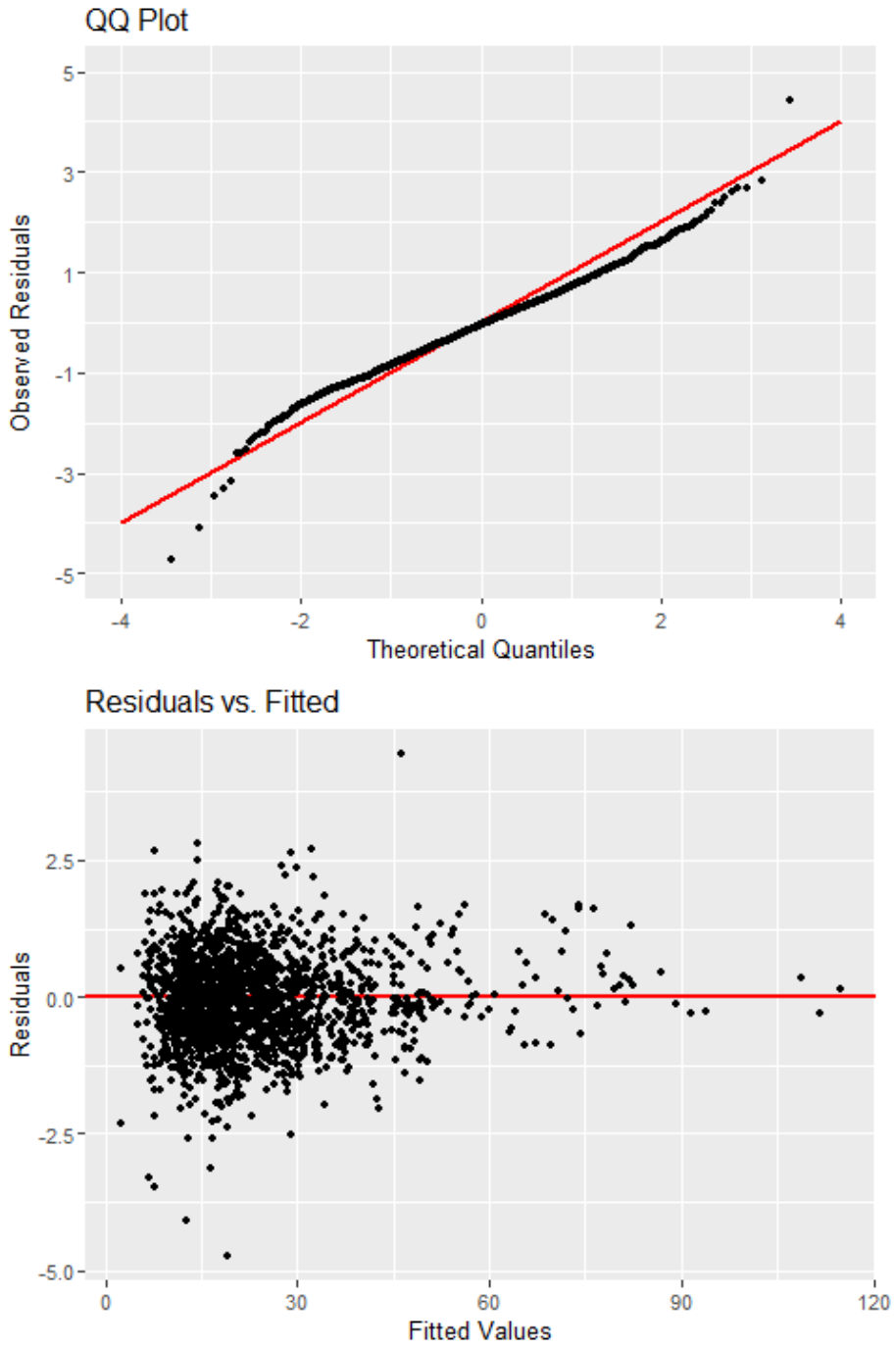

Figure 5: QQ plot and Residuals vs. Fitted plot for the finalised model, incorporating all selected covariates and interaction smooths, as well as the appropriate spatial, temporal and spatio-temporal effects.

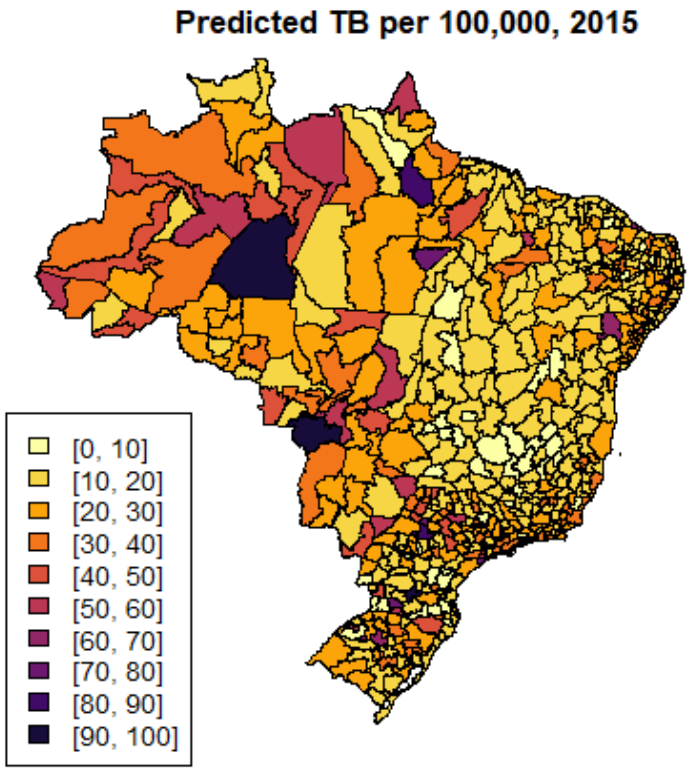

Figure 6: Predicted TB risk (counts per 100,000 population) for 2015 across the 557 microregions of Brazil.

Appendix B - Tables

Covariate	Minimum	Mean	Maximum	Variance	Skewness
Indigenous	0.010	0.843	50.646	12.463	9.930
Illiteracy	2.336	14.802	41.137	86.532	0.596
Urbanisation	22.336	71.961	99.927	273.173	-0.260
Density	0.422	0.621	1.675	0.028	2.709
Poverty	5.923	44.371	77.883	375.116	-0.003
Poor Sanitation	0.047	16.449	58.433	156.499	0.720
Unemployment	1.128	6.930	20.438	6.561	0.812
Timeliness	0.000	47.668	96.685	461.162	0.088
TB per 100,000	0.000	23.540	117.726	237.745	2.132

Table 1: Summary statistics (Mean, Variance, {Minimum, Maximum}, Skewness) for each socio-economic covariate, as well as the response variable TB per 100,000.

Covariate	Longitude	Latitude
Indigenous	-0.304	0.196
Illiteracy	0.475	0.690
Urbanisation	-0.148	-0.452
Density	-0.225	0.644
Poverty	0.377	0.779
Poor Sanitation	-0.003	0.591
Unemployment	0.318	0.494
Timeliness	-0.305	-0.355

Table 2: Pearson Correlation Coefficients (r) between each of the 8 socio-economic covariates and the longitude and latitude values for the centroid of the associated microregion.

p	GCV	Adj. r-squared	Explained Deviance	AIC
1.1	3.967	0.343	40.169	4199.670
1.2	2.886	0.341	40.253	4179.618
1.3	2.108	0.341	41.160	4167.913
1.4	1.545	0.338	40.213	4156.971
1.5	1.137	0.336	40.065	4148.664
1.6	0.842	0.334	39.799	4142.999
1.7	0.628	0.331	39.353	4141.334
1.8	0.476	0.329	38.548	4147.559
1.9	0.379	0.327	36.559	4180.834

Table 3: Calculated Generalised Cross-Validation (GCV), Adjusted r^2, Explained Deviance (%) and AIC metrics for a range of values for p: 1.1 - 1.9.

Appendix C - Commented Code

C.1 - Project Initialisation

```
# Package imports
library(knitr)
library(mgcv)
library(e1071)
library(corrplot)
library(forecast)
library(fields)
library(maps)
library(sp)
library(ggplot2)
library(patchwork)
library(RColorBrewer)
```

```
# Load the provided dataset and associated functions
load('datasets_project.RData')
```

C.2 - Initial Data Manipulation

```
# Create TB per 100,000 variable and store in TBdata
TB_per_100000 <- TBdata$TB / TBdata$Population * 100000
TBdata <- cbind(TBdata[, 1:11], TB_per_100000 = TB_per_100000, TBdata[, 12:ncol(TBdata)])

# Create epsilon-adjustment variable, for use in gamma family GAM
TB_per_100000_adjusted <- c()
for (i in 1:length(TBdata$TB)) {
  if (TBdata$TB[i] == 0) {
    TB_per_100000_adjusted <- c(TB_per_100000_adjusted, TBdata$TB_per_100000[i] + 0.01)
  } else {
    TB_per_100000_adjusted <- c(TB_per_100000_adjusted, TBdata$TB_per_100000[i])
  }
}
TBdata$TB_per_100000_adjusted <- TB_per_100000_adjusted

# Create subsets of the data frame for each response Year
TBdata_2012 <- TBdata[TBdata$Year == 2012, ]
TBdata_2013 <- TBdata[TBdata$Year == 2013, ]
TBdata_2014 <- TBdata[TBdata$Year == 2014, ]
```

C.3 - Asserting that Covariates are Consistent Across Years

```
# Check that 2012 covariates == 2013 covariates == 2014 covariates
table(unlist(TBdata_2012[, 1:8] == TBdata_2013[, 1:8]))
```

```
##
## TRUE
## 4456
```

```
table(unlist(TBdata_2013[, 1:8] == TBdata_2014[, 1:8]))
```

```
##
## TRUE
```

```
## 4456
```

C.4 - Producing Table 1

```r
# Calculate table of summary statistics
Covariate <- colnames(TBdata)[c(1:8, 12)]

Minimum <- c()
Mean <- c()
Maximum <- c()
Variance <- c()
Skewness <- c()

for (i in 1:length(Covariate)) {
  Minimum <- c(Minimum, min(TBdata[, Covariate[i]]))
  Mean <- c(Mean, mean(TBdata[, Covariate[i]]))
  Maximum <- c(Maximum, max(TBdata[, Covariate[i]]))
  Variance <- c(Variance, var(TBdata[, Covariate[i]]))
  # Calculates Fisher-Pearson standardised moment coefficient of skewness
  Skewness <- c(Skewness, skewness(TBdata[, Covariate[i]]))
}

# Store in a data frame and print via kable
exploratory_variables <- data.frame(Covariate, Minimum, Mean, Maximum, Variance, Skewness)
exploratory_variables[6, 1] <- "Poor Sanitation"
exploratory_variables[9, 1] <- "TB per 100,000"
kable(exploratory_variables, digits = 3)
```

C.5 - Producing Figure 1

```r
# Create plots with histograms of variable distribution
set_palette <- brewer.pal(n = 8, name = "Set2")
dark_palette <- brewer.pal(n = 8, name = "Dark2")

# Produce individual ggplot histogram for each covariate
hist_indigenous      <- ggplot(data = TBdata_2014, aes(x = Indigenous)) +
                        geom_histogram(fill = set_palette[1], color = dark_palette[1],
                                       alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Indigenous")
hist_illiteracy      <- ggplot(data = TBdata_2014, aes(x = Illiteracy)) +
                        geom_histogram(fill = set_palette[2], color = dark_palette[2],
                                       alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Illiteracy")
hist_urbanisation    <- ggplot(data = TBdata_2014, aes(x = Urbanisation)) +
                        geom_histogram(fill = set_palette[3], color = dark_palette[3],
                                       alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Urbanisation")
hist_density         <- ggplot(data = TBdata_2014, aes(x = Density)) +
                        geom_histogram(fill = set_palette[4], color = dark_palette[4],
                                       alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Density")
hist_poverty         <- ggplot(data = TBdata_2014, aes(x = Poverty)) +
                        geom_histogram(fill = set_palette[5], color = dark_palette[5],
```

```
                                    alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Poverty")
hist_poor_sanitation <- ggplot(data = TBdata_2014, aes(x = Poor_Sanitation)) +
                        geom_histogram(fill = set_palette[6], color = dark_palette[6],
                                    alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Poor Sanitation")
hist_unemployment    <- ggplot(data = TBdata_2014, aes(x = Unemployment)) +
                        geom_histogram(fill = set_palette[7], color = dark_palette[7],
                                    alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Unemployment")
hist_timeliness      <- ggplot(data = TBdata_2014, aes(x = Timeliness)) +
                        geom_histogram(fill = set_palette[8], color = dark_palette[8],
                                    alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "Timeliness")
hist_TB_per_100000   <- ggplot(data = TBdata_2014, aes(x = TB_per_100000)) +
                        geom_histogram(fill = "red", color = "darkred", alpha = 0.6) +
                        labs(x = NULL, y = NULL, title = "TB per 100,000")

# Define layout - bottom row has merged columns
hists_layout <- (hist_indigenous | hist_illiteracy) /
                (hist_urbanisation | hist_density) /
                (hist_poverty | hist_poor_sanitation) /
                (hist_unemployment | hist_timeliness) /
                hist_TB_per_100000

hists_layout <- hists_layout + plot_layout(ncol = 1, heights = c(1, 1, 1, 1, 2))
hists_layout
```

C.6 - Producing Figure 2

```
# Produce correlation matrix across 8 covariates
corr_matrix <- cor(TBdata_2014[, 1:8], method = "pearson")
# Convert negative R-squared valuess to positive
abs_corr_matrix <- abs(corr_matrix)

# Create a bidirectional palette (corrplot only uses top half by default)
palette_forward <- viridis(500, option = "B")
palette_backward <- viridis(500, option = "B", direction = -1)
palette_bidirectional <- c(palette_forward, palette_backward[2:500])

# Produce heatmap - only top half
corrplot(abs_corr_matrix, method = "color", type = "upper", col.lim = c(0, 1),
         col = palette_bidirectional, tl.col = "black", tl.cex = 0.8)

# Poverty - Illiteracy = 0.897
# Poor Sanitation - Urbanisation = 0.834
# Poor Sanitation - Poverty = 0.761
# Poverty - Urbanisation = 0.750
# Illiteracy - Urbanisation = 0.707
```

C.7 - Producing Figure 3

```
# Edit of the provided plot.map function
plot.map <- function(x,bin.width=15,main="",cex=1){
  # Now uses fixed bin widths, rather than fixed number of levels
  Q <- seq(0, ceiling(max(x)/10)*10, by = bin.width)
  n <- 557
  # Inferno colour palette used over Viridis
  cols <- viridis(length(Q), option = "B", direction = -1)
  col <- rep(cols[1],n)
  for(i in 2:length(Q)){
    col[x>=Q[i] & x<Q[i+1]] <- cols[i]
  }
  legend.names <- c()
  for(i in 1:length(Q)-1){
    legend.names[i] <- paste("[",round(Q[i],2),", ",round(Q[i+1],2),"]",sep="")
  }
  # Reduce outer whitespace
  par(oma = c(0, 0, 0, 0))
  # Added black borders to outline each microregion
  plot(brasil_micro,col=col,main="", border = "black", lwd = 0.5)
  # Microadjusted locations of title and legend
  title(main = main, line = 0)
  legend('bottomleft',legend=legend.names,fill=cols,cex=cex)
}

plot.map(TBdata_2014$TB_per_100000, bin.width = 15, main = "TB per 100,000, 2014")
```

C.8 - Producing Table 2

```
# List of socio-economic covariates
Covariate <- c("Indigenous", "Illiteracy", "Urbanisation", "Density",
               "Poverty", "Poor_Sanitation", "Unemployment", "Timeliness")

# Calculate r between each covariate and both longitude and latitude
Longitude <- c()
Latitude <- c()
for (i in 1:length(Covariate)) {
  Longitude <- c(Longitude, cor(TBdata_2014$lon, TBdata_2014[, Covariate[i]]))
  Latitude <- c(Latitude, cor(TBdata_2014$lat, TBdata_2014[, Covariate[i]]))
}

# Store in a dataframe and print via kabla
loc_corr <- data.frame(Covariate, Longitude, Latitude)
loc_corr[6, 1] <- "Poor Sanitation"
kable(loc_corr, digits = 3)
```

C.9 - Gamma and Tweedie Families, Initial Comparison

```
# Basic gamma model - all covariates w/ smoothing, link function = log
initial_gamma_model   <- gam(TB_per_100000_adjusted ~ s(Indigenous) + s(Illiteracy) +
                             s(Density) + s(Urbanisation) + s(Poor_Sanitation) +
                             s(Poverty) + s(Unemployment) + s(Timeliness),
```

```
                            data = TBdata_2014, family = Gamma(link = "log"))

summary(initial_gamma_model)

# Basic tweedie model - all covariates w/ smoothing, p = 1.5, link function = log
initial_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous) + s(Illiteracy) + s(Density) +
                            s(Urbanisation) + s(Poor_Sanitation) + s(Poverty) +
                            s(Unemployment) + s(Timeliness), data = TBdata_2014,
                            family = Tweedie(p = 1.5, link = power(0)))

summary(initial_tweedie_model)
```

C.10 - Producing Comparison Plots for Gamma and Tweedie Models

```
# Calculate residuals for Gamma model
dev_residuals_gamma <- residuals(initial_gamma_model, type = "deviance")
# Calculate theoretical quantiles for Gamma model
n <- length(dev_residuals_gamma)
theoretical_quantiles_gamma <- qnorm(ppoints(n))
dev_residuals_gamma <- sort(dev_residuals_gamma)
# Calculate residuals for Tweedie model
dev_residuals_tweedie <- residuals(initial_tweedie_model, type = "deviance")
# Calculate theoretical quantiles for Tweedie model
n <- length(dev_residuals_tweedie)
theoretical_quantiles_tweedie <- qnorm(ppoints(n))
dev_residuals_tweedie <- sort(dev_residuals_tweedie)

# Store in single data frame for the ggplots
QQ_compare_df <- data.frame(dev_residuals_gamma, theoretical_quantiles_gamma,
                            dev_residuals_tweedie, theoretical_quantiles_tweedie)

QQ_gamma <- ggplot(aes(x = theoretical_quantiles_gamma, y = dev_residuals_gamma),
                   data = QQ_compare_df) +
  geom_segment(aes(x = -4, y = -4, xend = 4, yend = 4), color = "red", linewidth = 0.5) +
  geom_point(size = 1) + labs(x = "Theoretical Quantiles", y = "Observed Residuals",
                      title = "QQ Plot - Gamma") +
  scale_x_continuous(limits = c(-4, 4), breaks = seq(-4, 4, by = 2)) +
  scale_y_continuous(limits = c(-5, 5), breaks = seq(-5, 5, by = 2))

QQ_tweedie <- ggplot(aes(x = theoretical_quantiles_tweedie, y = dev_residuals_tweedie),
                   data = QQ_compare_df) +
  geom_segment(aes(x = -4, y = -4, xend = 4, yend = 4), color = "red", linewidth = 0.5 ) +
  geom_point(size = 1) + labs(x = "Theoretical Quantiles", y = "Observed Residuals",
                      title = "QQ Plot - Tweedie") +
  scale_x_continuous(limits = c(-4, 4), breaks = seq(-4, 4, by = 2)) +
  scale_y_continuous(limits = c(-5, 5), breaks = seq(-5, 5, by = 2))

# Calculate fitteds and residuals for both Gamma and Tweedie models
gamma_fitted <- initial_gamma_model$fitted.values
gamma_residuals <- residuals(initial_gamma_model)
tweedie_fitted <- initial_tweedie_model$fitted.values
tweedie_residuals <- residuals(initial_tweedie_model)
```

```r
# Store in single data frame for the ggplots
resids_compare_df <- data.frame(gamma_fitted, gamma_residuals,
                                tweedie_fitted, tweedie_residuals)

resids_gamma <- ggplot(data = resids_compare_df,
                       aes(x = gamma_fitted, y = gamma_residuals)) +
  geom_hline(yintercept = 0, color = "red", linewidth = 1) +
  geom_point(size = 1) + labs(x = "Fitted Values", y = "Residuals",
                      title = "Residuals vs. Fitted - Gamma")

resids_tweedie <- ggplot(data = resids_compare_df,
                         aes(x = tweedie_fitted, y = tweedie_residuals)) +
  geom_hline(yintercept = 0, color = "red", linewidth = 1) +
  geom_point(size = 1) + labs(x = "Fitted Values", y = "Residuals",
                      title = "Residuals vs. Fitted - Tweedie")

# Define layout
model_compare_layout <- (QQ_gamma | QQ_tweedie) /
                        (resids_gamma | resids_tweedie)

model_compare_layout <- model_compare_layout +
                        plot_layout(ncol = 1, heights = c(1, 1))
# Print plot
model_compare_layout
```

C.11 - Testing for p - Producing Table 3

```r
# Calculate GCV, r^2, explained deviance and AIC across p values: 1.1 - 1.9
p <- c()
GCV <- c()
r_squared <- c()
Deviance_Explained <- c()
AIC <- c()

for (i in 11:19) {
  p_test_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous) + s(Illiteracy) + s(Density) +
                                s(Urbanisation) + s(Poor_Sanitation) + s(Poverty) +
                                s(Unemployment) + s(Timeliness), data = TBdata_2014,
                              family = Tweedie(p = i/10, link = power(0)))

  p <- c(p, i/10)
  GCV <- c(GCV, p_test_tweedie_model$gcv.ubre)
  r_squared <- c(r_squared, summary(p_test_tweedie_model)$r.sq)
  Deviance_Explained <- c(Deviance_Explained, summary(p_test_tweedie_model)$dev.expl * 100)
  AIC <- c(AIC, AIC(p_test_tweedie_model))
}

# Store all metrics in a data frame and print via kable
p_test <- data.frame(p, GCV, r_squared, Deviance_Explained, AIC)
kable(p_test, digits = 3)
```

C.12 - Model Initialisation, selected p = 1.3

```
# Produce initial model with selected p value = 1.3
p_test_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous) + s(Illiteracy) + s(Density) +
                              s(Urbanisation) + s(Poor_Sanitation) + s(Poverty) +
                              s(Unemployment) + s(Timeliness), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(p_test_tweedie_model)
# s(Illiteracy)      1.000  1.000  0.015 0.902986
# s(Urbanisation)    1.000  1.000  2.883 0.090117 .
# s(Poverty)         2.924  3.725  0.584 0.570893

# R-sq.(adj) =  0.341   Deviance explained = 41.2%
# GCV = 2.1084  Scale est. = 2.3134    n = 557

AIC(p_test_tweedie_model)
# 4167.913
```

C.13 - Model Improvement Process

```
# Remove Illiteracy, Poverty, Urbanisation - add as tensor due to high correlation
test_tweedie_model_1 <- gam(TB_per_100000 ~ s(Indigenous) + s(Density) +
                              s(Poor_Sanitation) + s(Unemployment) + s(Timeliness) +
                              te(Illiteracy, Poverty, Urbanisation), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(test_tweedie_model_1)
# s(Indigenous)                       1.691  2.033 2.664 0.067688 .

# R-sq.(adj) =  0.384   Deviance explained = 46.6%
# GCV = 2.0304  Scale est. = 2.1057    n = 557

k.check(test_tweedie_model_1)
# All p values 0.05 and above

AIC(test_tweedie_model_1)
# 4142.198


# Perform BoxCox transformation on Indigenous due to extreme skew
lambda_Indigenous <- BoxCox.lambda(TBdata_2014$Indigenous)
Indigenous_BoxCox <- BoxCox(TBdata_2014$Indigenous, lambda_Indigenous)
TBdata_2014 <- cbind(TBdata_2014, Indigenous_BoxCox)

test_tweedie_model_2 <- gam(TB_per_100000 ~ s(Indigenous_BoxCox) + s(Density) +
                              s(Poor_Sanitation) + s(Unemployment) + s(Timeliness) +
                              te(Illiteracy, Poverty, Urbanisation), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(test_tweedie_model_2)
# s(Indigenous)                       1.691  2.033 2.664 0.067688 .

# R-sq.(adj) =  0.395   Deviance explained = 47.3%
```

```r
# GCV = 2.0034  Scale est. = 2.0832    n = 557

k.check(test_tweedie_model_2)
# All p values 0.05 and above

AIC(test_tweedie_model_2)
# 4134.614

# Add ti() smooth between Density and Poor_Sanitation (context reasons)
test_tweedie_model_3 <- gam(TB_per_100000 ~ s(Indigenous_BoxCox) + s(Density) +
                              s(Poor_Sanitation) + s(Unemployment) + s(Timeliness) +
                              te(Illiteracy, Poverty, Urbanisation) +
                              ti(Density, Poor_Sanitation), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(test_tweedie_model_3)
# s(Poor_Sanitation)                  3.734  4.738  2.085 0.065215 .
# ti(Density,Poor_Sanitation)         3.787  4.707  1.876 0.081836 .

# R-sq.(adj) =  0.417   Deviance explained =   49%
# GCV = 1.9779  Scale est. = 2.0095    n = 557

k.check(test_tweedie_model_3)
# s(Timeliness)                        9  3.081716 0.9203576  0.0400

AIC(test_tweedie_model_3)
# 4125.289

# Increase k for Indigenous and Timeliness due to prior k.check issues
test_tweedie_model_4 <- gam(TB_per_100000 ~ s(Indigenous_BoxCox, k=50) + s(Density) +
                              s(Poor_Sanitation) + s(Unemployment) + s(Timeliness, k=50) +
                              te(Illiteracy, Poverty, Urbanisation) +
                              ti(Density, Poor_Sanitation), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(test_tweedie_model_4)
# s(Poor_Sanitation)                  2.891  3.732 1.633  0.16357

# R-sq.(adj) =  0.421   Deviance explained = 50.1%
# GCV =  1.989  Scale est. = 1.9582    n = 557

k.check(test_tweedie_model_4)
# s(Indigenous_BoxCox)                49  1.000007 0.8708271  0.0025

# Add ti() smooth between Indigenous and Unemployment (context reasons)
test_tweedie_model_5 <- gam(TB_per_100000 ~ s(Indigenous_BoxCox, k=50) + s(Density) +
                              s(Poor_Sanitation) + s(Unemployment) + s(Timeliness, k=50) +
                              te(Illiteracy, Poverty, Urbanisation) +
                              ti(Density, Poor_Sanitation) +
                              ti(Indigenous_BoxCox, Unemployment), data = TBdata_2014,
                            family = Tweedie(p = 1.3, link = power(0)))

summary(test_tweedie_model_5)
```

```
# s(Poor_Sanitation)                    3.494  4.446 1.947  0.08763 .

# R-sq.(adj) =  0.445   Deviance explained = 51.4%
# GCV = 1.9703  Scale est. = 1.9195    n = 557

k.check(test_tweedie_model_5)
# s(Indigenous_BoxCox)                  49  1.000344 0.8802033  0.0025

AIC(test_tweedie_model_5)
# 4119.882
```

C.14 - Incorporating Spatial, Temporal and Spatio-Temporal Effects

```
# Switch to all-year dataset
# Perform BoxCox transformation on all-year TBdata set
lambda_Indigenous <- BoxCox.lambda(TBdata$Indigenous)
Indigenous_BoxCox <- BoxCox(TBdata$Indigenous, lambda_Indigenous)
TBdata <- cbind(TBdata, Indigenous_BoxCox)

# Add spatial, temporal and spatio-temporal effects
test_final_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous_BoxCox, k=50) + s(Density) +
                                  s(Poor_Sanitation) + s(Unemployment) + s(Timeliness, k=50) +
                                  te(Illiteracy, Poverty, Urbanisation) +
                                  ti(Density, Poor_Sanitation) +
                                  ti(Indigenous_BoxCox, Unemployment) +
                                  s(lat, lon, bs = "tp") + # Spatial
                                  s(Year, bs = "cr", k = 3) + # Temporal
                                  # Spatio-Temporal
                                  te(lat, lon, Year, bs=c("tp", "tp", "cr"), k=c(10, 10, 3)),
                                data = TBdata, # Adjusted to all-year dataset
                                family = Tweedie(p = 1.3, link = power(0)))

summary(test_final_tweedie_model)
# s(Indigenous_BoxCox)                 45.949  48.194  4.328  < 2e-16 ***
# s(Density)                            8.730   8.939  7.240  < 2e-16 ***
# s(Poor_Sanitation)                    8.168   8.719  2.617 0.008448 **
# s(Unemployment)                       5.171   6.268 11.172  < 2e-16 ***
# s(Timeliness)                        40.350  45.130  4.613  < 2e-16 ***
# te(Illiteracy,Poverty,Urbanisation) 113.687 115.177  6.175  < 2e-16 ***
# ti(Density,Poor_Sanitation)          12.521  13.408  8.473  < 2e-16 ***
# ti(Indigenous_BoxCox,Unemployment)    9.331  10.648  2.969 0.000978 ***
# s(lat,lon)                           26.846  27.695  3.961  < 2e-16 ***
# s(Year)                               1.646   1.875  0.813 0.430805
# te(lat,lon,Year)                     68.605  78.850  1.440 0.008242 **

# R-sq.(adj) =  0.781   Deviance explained = 78.9%
# GCV = 1.0801  Scale est. = 0.8315    n = 1671

AIC(test_final_tweedie_model)
# 11312.15


# Remove s(Year) as it is an insignificant covariate
```

```r
final_tweedie_model <- gam(TB_per_100000 ~ s(Indigenous_BoxCox, k=50) + s(Density) +
                             s(Poor_Sanitation) + s(Unemployment) + s(Timeliness, k=50) +
                             te(Illiteracy, Poverty, Urbanisation) +
                             ti(Density, Poor_Sanitation) +
                             ti(Indigenous_BoxCox, Unemployment) +
                             s(lat, lon, bs = "tp") +
                             te(lat, lon, Year, bs = c("tp", "tp", "cr"), k = c(10, 10, 3)),
                           data = TBdata,
                           family = Tweedie(p = 1.3, link = power(0)))

summary(final_tweedie_model)
# s(Indigenous_BoxCox)                 46.257  48.316  4.338  < 2e-16 ***
# s(Density)                            8.562   8.893  7.189  < 2e-16 ***
# s(Poor_Sanitation)                    8.106   8.689  2.608 0.008060 **
# s(Unemployment)                       4.683   5.751 11.604  < 2e-16 ***
# s(Timeliness)                        40.429  45.195  4.713  < 2e-16 ***
# te(Illiteracy,Poverty,Urbanisation) 111.425 112.902  6.187  < 2e-16 ***
# ti(Density,Poor_Sanitation)          12.398  13.318  8.444  < 2e-16 ***
# ti(Indigenous_BoxCox,Unemployment)    8.684   9.939  3.090 0.000916 ***
# s(lat,lon)                           27.924  28.385  3.819  < 2e-16 ***
# te(lat,lon,Year)                     69.185  79.641  1.518 0.002807 **

# R-sq.(adj) =   0.78   Deviance explained = 78.7%
# GCV = 1.0828  Scale est. = 0.83666   n = 1671

AIC(final_tweedie_model)
# 11317.9
```

C.15 - Producing Figure 5

```r
# Calculate residuals for Final model
dev_residuals_tweedie <- residuals(final_tweedie_model, type = "deviance")
# Calculate theoretical quantiles for Final model
n <- length(dev_residuals_tweedie)
theoretical_quantiles_tweedie <- qnorm(ppoints(n))
dev_residuals_tweedie <- sort(dev_residuals_tweedie)

# Store in single data frame for the ggplot
QQ_compare_df <- data.frame(dev_residuals_tweedie, theoretical_quantiles_tweedie)

QQ_tweedie <- ggplot(aes(x = theoretical_quantiles_tweedie, y = dev_residuals_tweedie),
                     data = QQ_compare_df) +
  geom_segment(aes(x = -4, y = -4, xend = 4, yend = 4), color = "red", linewidth = 1) +
  geom_point(size = 1.2) + labs(x = "Theoretical Quantiles", y = "Observed Residuals",
                        title = "QQ Plot") +
  scale_x_continuous(limits = c(-4, 4), breaks = seq(-4, 4, by = 2)) +
  scale_y_continuous(limits = c(-5, 5), breaks = seq(-5, 5, by = 2))

# Calculate fitteds and residuals for the Final model
tweedie_fitted <- final_tweedie_model$fitted.values
tweedie_residuals <- residuals(final_tweedie_model)

# Store in single data frame for the ggplots
```

```r
resids_compare_df <- data.frame(tweedie_fitted, tweedie_residuals)

resids_tweedie <- ggplot(data = resids_compare_df,
                         aes(x = tweedie_fitted, y = tweedie_residuals)) +
  geom_hline(yintercept = 0, color = "red", linewidth = 1) +
  geom_point(size = 1.2) + labs(x = "Fitted Values", y = "Residuals",
                       title = "Residuals vs. Fitted")

# Define layout
model_compare_layout <- (QQ_tweedie) /
                        (resids_tweedie)

model_compare_layout <- model_compare_layout +
                        plot_layout(ncol = 1, heights = c(2))
# Print plot
model_compare_layout
```

C.16 - Producing Figure 6

```r
# Produce dataset for 2015 (under assumption the covariate values do *not* change)
prediction_df <- data.frame(
  Indigenous_BoxCox = TBdata_2014$Indigenous_BoxCox,   # Predictor values for the region
  Density = TBdata_2014$Density,              # Predictor values
  Poor_Sanitation = TBdata_2014$Poor_Sanitation,     # Predictor values
  Unemployment = TBdata_2014$Unemployment,         # Predictor values
  Timeliness = TBdata_2014$Timeliness,           # Predictor values
  Illiteracy = TBdata_2014$Illiteracy,           # Predictor values
  Poverty = TBdata_2014$Poverty,              # Predictor values
  Urbanisation = TBdata_2014$Urbanisation,         # Predictor values
  lat = TBdata_2014$lat,                  # Latitude
  lon = TBdata_2014$lon,                  # Longitude
  Year = rep(2015, times = 557),                  # Year
  Region = TBdata_2014$Region      # Region identifier
)

# Create predicted values
preds <- predict(final_tweedie_model, newdata = prediction_df, type = "response")
plot.map(preds, bin.width = 10, main = "Predicted TB per 100,000, 2015")
```

C.17 - Asserting Microregions with Highest Risk for 2015

```r
# Add predicted values to a dataframe and sort by highest
prediction_df <- cbind(prediction_df, preds)
sorted_prediction_df <- prediction_df[order(-prediction_df$preds),]

# Visualise the regions with highest risk
head(sorted_prediction_df, 10)
#     Indigenous_BoxCox  .. Region    preds
# 20         -0.4351573  ...  13007 95.62862
# 533        -1.3571425  ...  51017 94.98226
# 411        -2.1437823  ...  35063 93.70354
# 191        -3.1852758  ...  26016 87.34831
# 37         -2.0717111  ...  15007 80.63534
```

```
# 404        -0.6517667  ...  35056 80.45733
# 383        -3.6362549  ...  35035 79.67354
# 38         -2.9829847  ...  15008 77.05469
# 348        -2.6983294  ...  33018 75.01468
# 449        -1.6994899  ...  41038 71.90962
```

Appendix D - AI Declaration

AI-supported use is permitted in this assessment. I acknowledge the following uses of GenAI tools in this assessment:

[ ] I have used GenAI tools for developing ideas.

[ ] I have used GenAI tools to assist with research or gathering information.

[ ] I have used GenAI tools to help me understand key theories and concepts.

[ ] I have used GenAI tools to identify trends and themes as part of my data analysis.

[ ] I have used GenAI tools to suggest a plan or structure for my assessment.

[ ] I have used GenAI tools to give me feedback on a draft.

[$\surd$] I have used GenAI tool to generate images, figures or diagrams.

[ ] I have used GenAI tools to proofread and correct grammar or spelling errors.

[ ] I have used GenAI tools to generate citations or references.

[ ] Other [please specify]/.

[ ] I have not used any GenAI tools in preparing this assessment.

I declare that I have referenced use of GenAI outputs within my assessment in line with the University referencing guidelines.