Group Project Report

2023-03-19

The Model

As the target variable is the count of TB cases, the basic form the model which is designed to explain the ratio of TB cases per capita has the form

$$TB_{i,t} \sim Pois(\eta_{i,t})$$

$$\log(\eta_{i,t}) = \log(Population_{i,t}) + \sum_{j=1}^{8} f_j(x_{i,t,j})$$

where $x_{i,t,j}$ for $j \in 1, ..., 8$ is the value one of the socio-economic variables mentioned in the task-description for year t. Having this model the model coefficients are explaining the relation between the explaining variables and the ratio of TB cases per capita:

$$\beta_0 + \sum_{j=1}^{8} \beta_j f_j(x_{i,j}) = \log(E(TB_{i,t}|\mathbf{x}_i, Population_{i,t})) - \log(Population_{i,t}) \ = \log\left(\frac{E(TB_{i,t}|\mathbf{x}_{i,t}, Population_{i,t})}{Population_{i,t}}\right)$$

Looking at the distribution of the residuals of the model one sees that the Poisson model, which has a fixed dispersion parameter is clearly overdispersed, and even with 60 knots per smooth term the model doesn't seem to have enough flexibility which may be another indicator that a Poisson model is overdispersed. Furthermore, the AIC drops clearly when fitting a Negative Binomial instead of a Poisson model. When having a look at the relationship between the squared residuals and the fitted values one sees that the relation is not exactly linear but rather linear than quadratic which would reflect the relation between model variance and the expected value in a Gamma Distribution Model (Additional evidence is provided by the Residuals vs. Fitted plot.). So the model distribution is changed to Negative Binomial with the same parameterization except for the feature that the count of TB cases is now Negative Binomial distributed with mean η_i as described above.

Given the ground model we investigate whether all given socio-economic variables are needed to explain the ratio of TB cases per capita or whether there is a less complex model. There for we try to drop the variables with the highest p-values for the hypothesis test $\beta_j = 0$ and perform an LRT to see whether the reduced model is as good as the more complex model. Leaving one variable out is repeated until the reduced model is significantly worse than more complex model. Dropping the Illiteracy variable does not make the model significantly worse. Next, the Poverty variable which has the 2nd lowest p-value for $\beta_j = 0$ in the initial model is dropped aditionally, but then the null hypothesis of the LRT that this model is as good as the model which only leaves out Illiteracy can be rejected at 5%-level. So in the following we use a model with all of the socio-economic variables except for Illiteracy.

Further extensions to the model can be reached by including 1) time, 2) space, or 3) both.

The temporal model changes the expression for η_i as follows:

$$\log(\eta_i) = \log(Population_{i,t}) + \beta_0 + \sum_{t=2012}^{2014} \sum_{j=1}^{7} \beta_{t,j} f_{t,j}(x_{i,t,j})$$

where $x_{i,t,j}$ is the value of the variable index by j for year t. For this model the AIC does not drop compared to the model which does not consider time.

The spatial model adds a smoothed term which is function of the longitude and the latitude. A bivariate function is used because it makes sense to assume that there are more cases at certain locations (defined by the interaction between latitude and longitude) than others, rather than that there are more cases at locations with a certain longitude for any latitude, or the other way round.

$$\log(\eta_i) = \log(Population_{i,t}) + \beta_0 + \sum_{j=1}^{7} \beta_j f_j(x_{i,t,j}) + \beta_8 f_8(lon_{i,t}, lat_{i,t})$$

Furthermore, there is a model which includes the term for the location and estimates a functional relation for each year and each explaining variable. The AIC of this model does not drop compared to the spatial model, so the spatial model is - given that it is simpler - the model which explains the ratio of TB cases per capita best.

$$\log(\eta_i) = \log(Population_{i,t}) + \beta_0 + \sum_{t=2012}^{2014} \left(\sum_{j=1}^{7} \beta_{t,j} f_{t,j}(x_{i,t,j}) + \beta_{s,t} f_{s,t}(lon_{i,t}, lat_{i,t}) \right)$$

Let us know have a look at the fit of the spatial model: It fits well even though the largest residuals are higher than expected from the model distribution. For districts that have a high number of cases the predictor does not seem as accurate as it should. But the highest residuals do not arise when the ratio of TB cases per capita is extraordinarily high, but rather when the absolute number of TB cases is high (see residuals vs. response). The variance of the model stil seems too low for those values given that there are some predicted values in that high segment of response values (absolut number of TB cases) where the prediction for the response value is lower than the actual value, and some where the prediction of the actual value is higher than the actual value.

Code

```
library (mgcv)
```

```
## Loading required package: nlme
```

```
## This is mgcv 1.8-42. For overview type 'help("mgcv-package")'.
```

```
par(mfrow = c(2,2))
#fit poisson model with socio-economic variables
model_poisson <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous) +  s(Illiteracy) +  s(Urbani
summary(model_poisson)
```

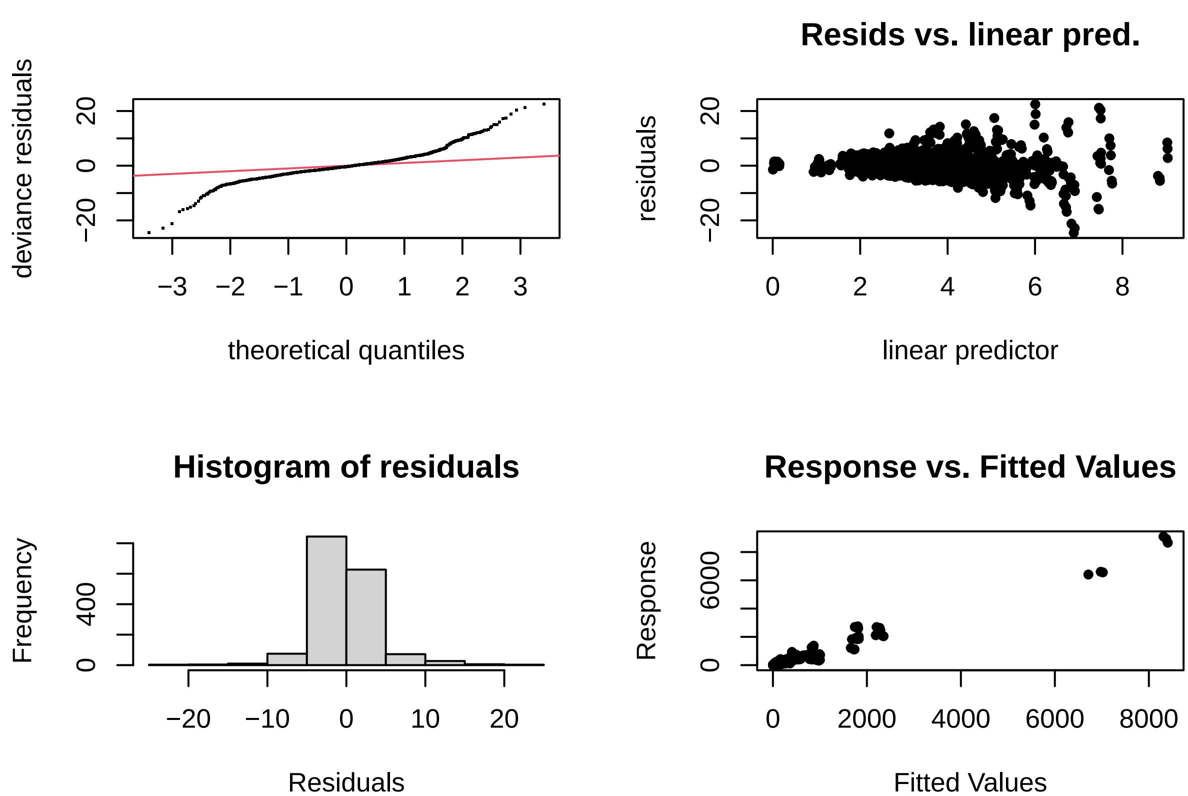
```
##
## Family: poisson
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous) + s(Illiteracy) +
##     s(Urbanisation) + s(Density) + s(Poverty) + s(Poor_Sanitation) +
##     s(Unemployment) + s(Timeliness)
##
## Parametric coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.449827   0.004199   -2012   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                        edf Ref.df Chi.sq p-value
```

```
## s(Indigenous)       8.961  8.999  569.4  <2e-16 ***
## s(Illiteracy)       8.989  9.000 2704.0  <2e-16 ***
## s(Urbanisation)     8.900  8.996 1490.4  <2e-16 ***
## s(Density)          8.985  9.000 1758.4  <2e-16 ***
## s(Poverty)          8.956  8.999 1470.2  <2e-16 ***
## s(Poor_Sanitation) 8.979  9.000 1327.0  <2e-16 ***
## s(Unemployment)    8.993  9.000 2423.5  <2e-16 ***
## s(Timeliness)       8.352  8.864  600.7  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.976   Deviance explained = 66.9%
## UBRE = 13.899  Scale est. = 1         n = 1671
```

```
model_poisson$aic
```

```
## [1] 34047.36
```

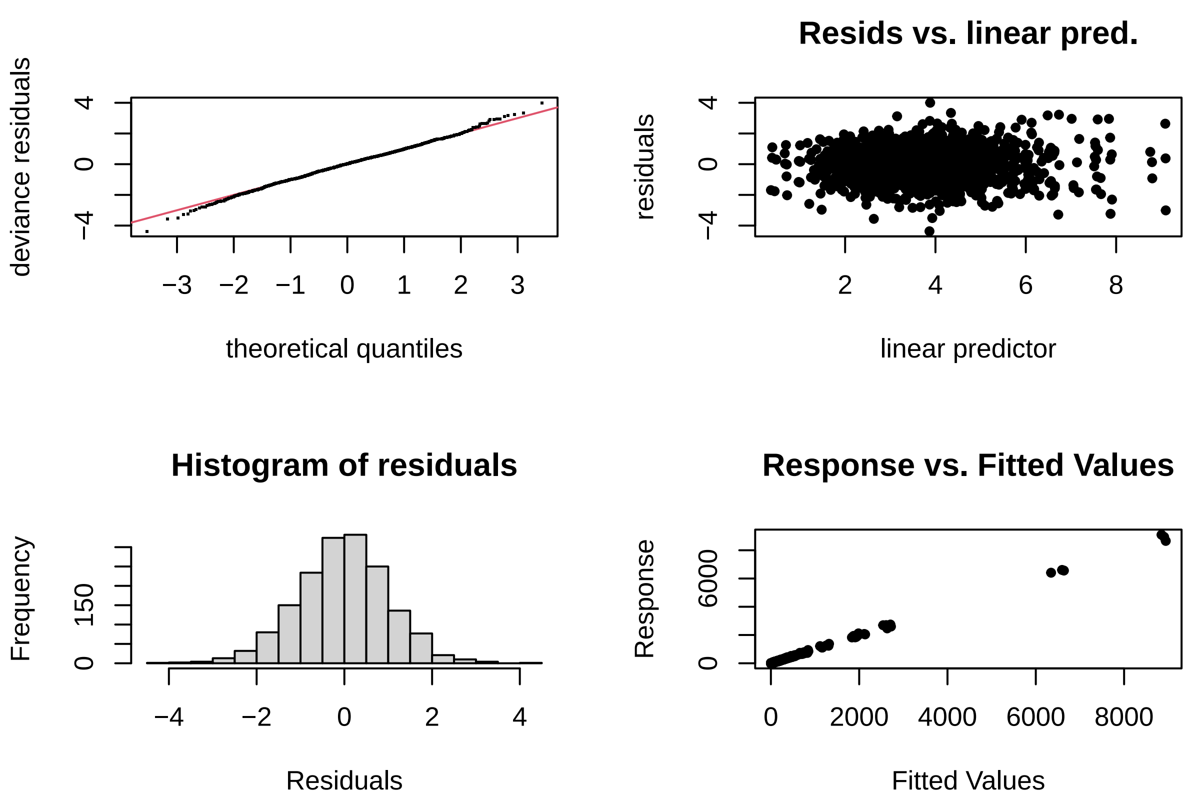
```
par(mfrow = c(2,2),pch = 20)
gam.check(model_poisson)
```




```
##
## Method: UBRE   Optimizer: outer newton
## full convergence after 13 iterations.
## Gradient range [2.754533e-08,1.177865e-05]
## (score 13.89908 & scale 1).
## Hessian positive definite, eigenvalue range [6.78691e-06,0.0006391723].
## Model rank =  73 / 73
```

```
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                       k'  edf k-index p-value
## s(Indigenous)       9.00 8.96    0.39  <2e-16 ***
## s(Illiteracy)       9.00 8.99    0.41  <2e-16 ***
## s(Urbanisation)     9.00 8.90    0.41  <2e-16 ***
## s(Density)          9.00 8.98    0.39  <2e-16 ***
## s(Poverty)          9.00 8.96    0.39  <2e-16 ***
## s(Poor_Sanitation) 9.00 8.98    0.40  <2e-16 ***
## s(Unemployment)     9.00 8.99    0.39  <2e-16 ***
## s(Timeliness)       9.00 8.35    0.43  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#add flexibility
model_poisson <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous, k = 80) +  s(Illiteracy, k =
gam.check(model_poisson)
```



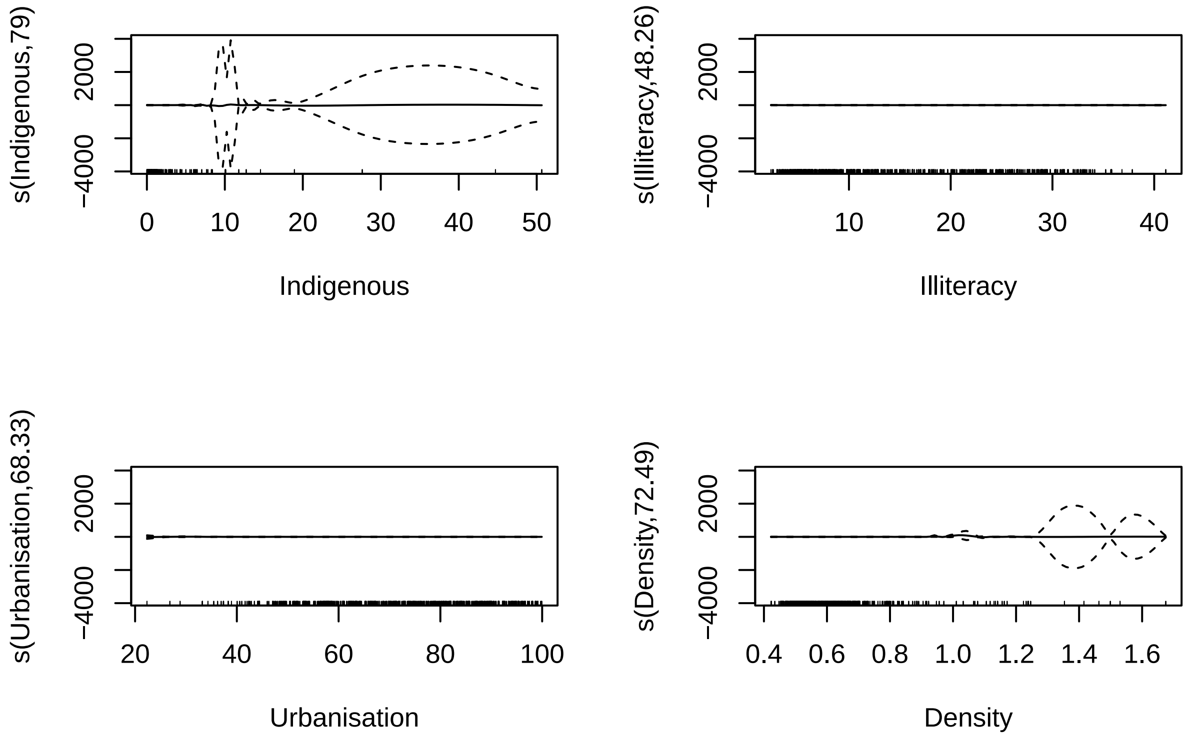

```
##
## Method: UBRE   Optimizer: outer newton
## full convergence after 20 iterations.
## Gradient range [1.19629e-08,6.409422e-07]
## (score 0.7355416 & scale 1).
## Hessian positive definite, eigenvalue range [2.050125e-08,0.001074078].
## Model rank =  633 / 633
```

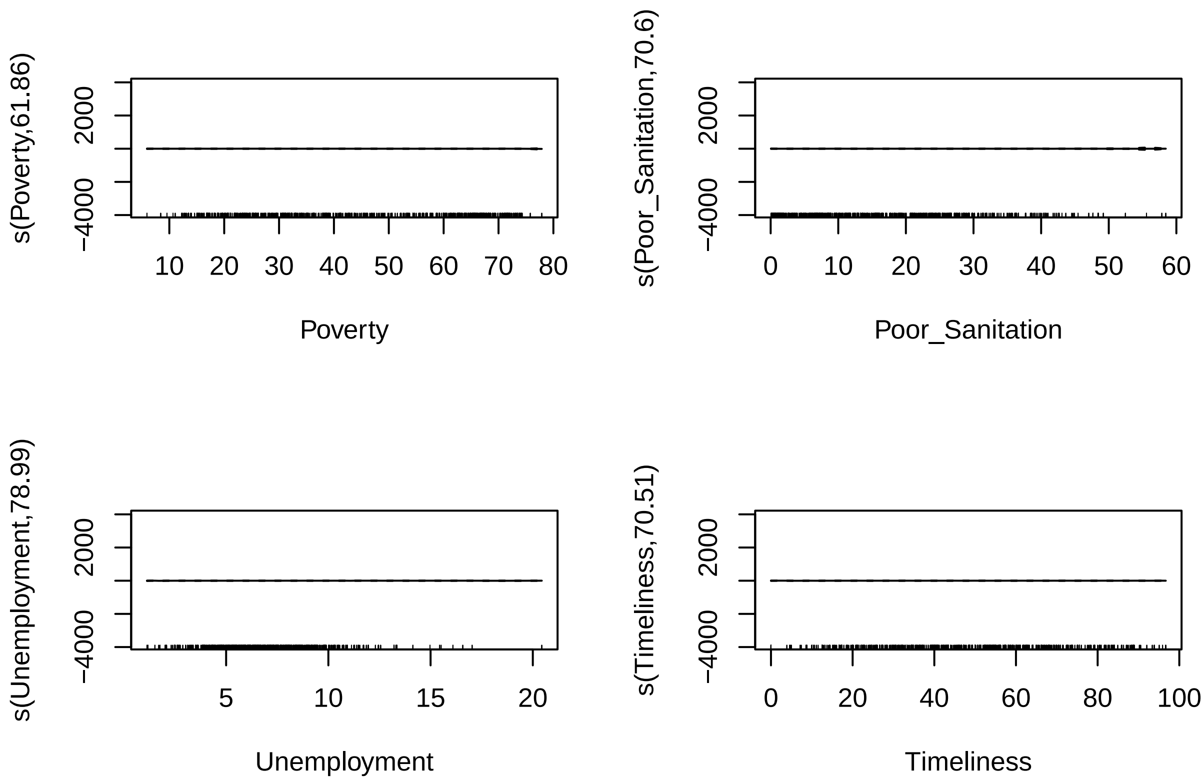
```
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                       k'  edf k-index p-value
## s(Indigenous)       79.0 79.0    1.34       1
## s(Illiteracy)       79.0 48.3    1.30       1
## s(Urbanisation)     79.0 68.3    1.32       1
## s(Density)          79.0 72.5    1.35       1
## s(Poverty)          79.0 61.9    1.30       1
## s(Poor_Sanitation) 79.0 70.6    1.32       1
## s(Unemployment)     79.0 79.0    1.31       1
## s(Timeliness)       79.0 70.5    1.26       1
```

```
summary(model_poisson) # SIGNS OF OVERFIT
```

```
##
## Family: poisson
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous, k = 80) + s(Illiteracy,
##     k = 80) + s(Urbanisation, k = 80) + s(Density, k = 80) +
##     s(Poverty, k = 80) + s(Poor_Sanitation, k = 80) + s(Unemployment,
##     k = 80) + s(Timeliness, k = 80)
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.512131   0.005417   -1571   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                       edf Ref.df Chi.sq p-value
## s(Indigenous)       79.00  79.00 324.42  <2e-16 ***
## s(Illiteracy)       48.26  49.65  58.57   0.176
## s(Urbanisation)     68.33  69.25 371.79  <2e-16 ***
## s(Density)          72.49  72.55 308.40  <2e-16 ***
## s(Poverty)          61.86  63.02 147.89  <2e-16 ***
## s(Poor_Sanitation) 70.60  71.04 254.68  <2e-16 ***
## s(Unemployment)     78.99  78.99 495.18  <2e-16 ***
## s(Timeliness)       70.51  71.53 470.06  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.999   Deviance explained = 97.6%
## UBRE = 0.73554  Scale est. = 1         n = 1671
```

```
plot(model_poisson) # SIGNS OF OVERFIT
```


s(Indigenous,79)
2000
−4000
0
10
20
30
40
50
Indigenous
s(Illiteracy,48.26)
2000
−4000
10
20
30
40
Illiteracy
s(Urbanisation,68.33)
2000
−4000
20
40
60
80
100
Urbanisation
s(Density,72.49)
2000
−4000
0.4
0.6
0.8
1.0
1.2
1.4
1.6
Density




```r
#fit negative binomial model with  socioeconomic
model_nb <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous) +  s(Illiteracy) + s(Urbanisatio
summary(model_nb)
```

```
##
## Family: Negative Binomial(6.146)
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous) + s(Illiteracy) +
##     s(Urbanisation) + s(Density) + s(Poverty) + s(Poor_Sanitation) +
##     s(Unemployment) + s(Timeliness)
##
## Parametric coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.42871    0.01094  -770.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                       edf Ref.df  Chi.sq  p-value
## s(Indigenous)       1.489  1.795  20.396 2.92e-05 ***
## s(Illiteracy)       1.008  1.017   0.246  0.63129
## s(Urbanisation)     6.634  7.773  24.089  0.00148 **
## s(Density)          4.579  5.672 132.693  < 2e-16 ***
## s(Poverty)          5.733  6.911  17.934  0.01516 *
```

```
## s(Poor_Sanitation) 6.123  7.297  73.103  < 2e-16 ***
## s(Unemployment)    5.798  7.000  62.050  < 2e-16 ***
## s(Timeliness)      4.101  5.097  64.474  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.861   Deviance explained = 43.9%
## -REML = 7237.2  Scale est. = 1         n = 1671
```

```
model_nb$aic
```

```
## [1] 14391.19
```

```
#fit a linear relation between sqaured residuals and prediction to see whether another model describes
summary(lm(log(model_nb$residuals^2) ~ log(predict(model_nb, type = 'response'))))
```

```
##
## Call:
## lm(formula = log(model_nb$residuals^2) ~ log(predict(model_nb,
##     type = "response")))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.2857  -1.2046   0.3557   1.5688   4.7149
##
## Coefficients:
##                                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                                -2.50894    0.19211 -13.060   <2e-16
## log(predict(model_nb, type = "response")) -0.10177    0.04902  -2.076    0.038
##
## (Intercept)                                ***
## log(predict(model_nb, type = "response")) *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.342 on 1669 degrees of freedom
## Multiple R-squared:  0.002576,   Adjusted R-squared:  0.001978
## F-statistic:  4.31 on 1 and 1669 DF,  p-value: 0.03803
```

```
#drop Illiteracy
model_nb_2 <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous) +  s(Urbanisation) + s(Density)


#LRT
anova.gam(model_nb_2, model_nb, test = 'LRT')
```

```
## Analysis of Deviance Table
##
## Model 1: TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness)
## Model 2: TB ~ offset(log(Population)) + s(Indigenous) + s(Illiteracy) +
##     s(Urbanisation) + s(Density) + s(Poverty) + s(Poor_Sanitation) +
##     s(Unemployment) + s(Timeliness)
##   Resid. Df Resid. Dev      Df Deviance Pr(>Chi)
## 1    1621.3      14319
```

```
## 2    1620.3      14318 0.99721  0.34543   0.5556
```

```r
#Null hypothesis not rejected -> drop poverty
model_nb_3 <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous) +  s(Urbanisation) + s(Density)
#LRT
anova.gam(model_nb_3, model_nb_2, test = 'LRT')
```

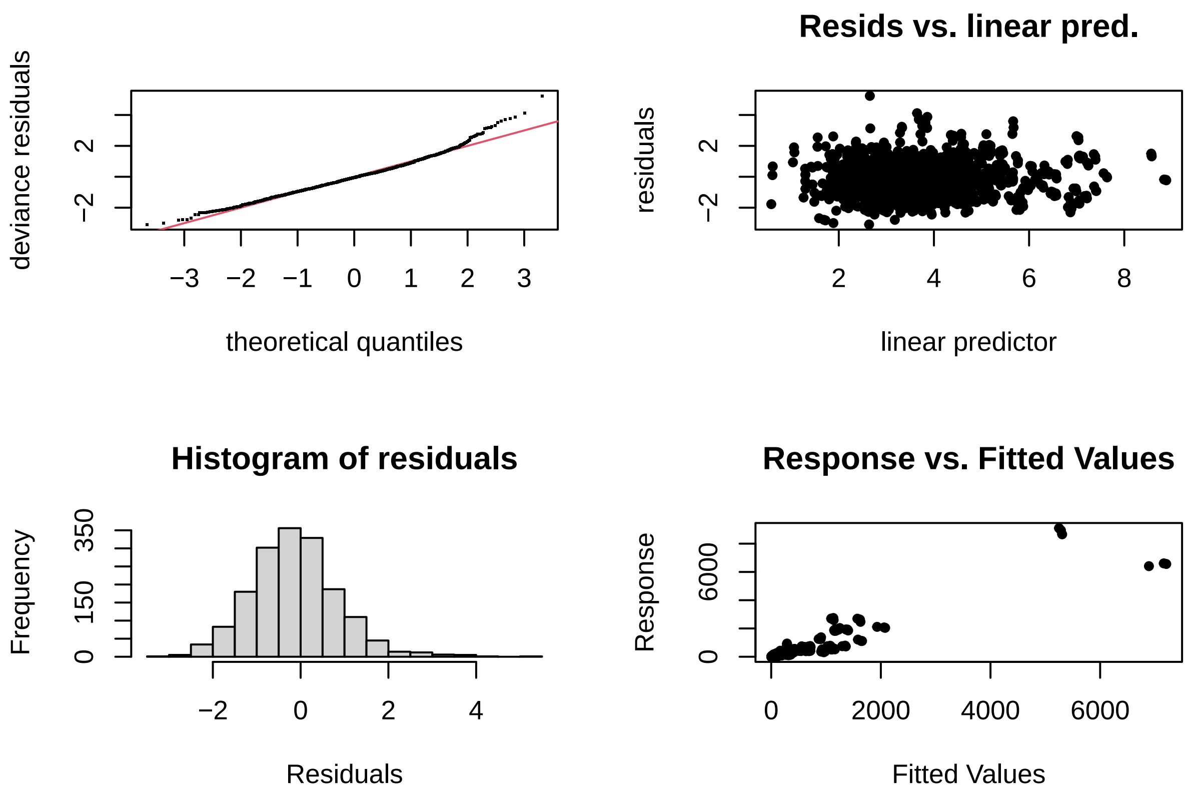
```
## Analysis of Deviance Table
##
## Model 1: TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poor_Sanitation) + s(Unemployment) + s(Timeliness)
## Model 2: TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness)
##   Resid. Df Resid. Dev     Df Deviance Pr(>Chi)
## 1    1630.5      14345
## 2    1621.3      14319 9.1169   26.444 0.001861 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# NULL hypothesis rejected -> drop only illiteracy, not poverty

model_nb_final <- model_nb_2
summary(model_nb_final)
```

```
##
## Family: Negative Binomial(6.146)
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.42863    0.01094  -770.6   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                      edf Ref.df Chi.sq  p-value
## s(Indigenous)      1.518  1.833  21.13 2.08e-05 ***
## s(Urbanisation)    6.610  7.752  23.73  0.00167 **
## s(Density)         4.578  5.667 147.64  < 2e-16 ***
## s(Poverty)         5.771  6.945  21.36  0.00394 **
## s(Poor_Sanitation) 6.119  7.293  76.07  < 2e-16 ***
## s(Unemployment)    5.776  6.977  64.21  < 2e-16 ***
## s(Timeliness)      4.106  5.103  66.42  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =   0.86   Deviance explained = 43.9%
## -REML = 7234.9  Scale est. = 1         n = 1671
```

```r
gam.check(model_nb_final)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 6 iterations.
## Gradient range [-5.66156e-06,7.265887e-06]
## (score 7234.878 & scale 1).
## Hessian positive definite, eigenvalue range [0.1154615,588.9389].
## Model rank =  64 / 64
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                        k'  edf k-index p-value
## s(Indigenous)       9.00 1.52    0.49  <2e-16 ***
## s(Urbanisation)     9.00 6.61    0.50  <2e-16 ***
## s(Density)          9.00 4.58    0.50  <2e-16 ***
## s(Poverty)          9.00 5.77    0.49  <2e-16 ***
## s(Poor_Sanitation) 9.00 6.12    0.50  <2e-16 ***
## s(Unemployment)     9.00 5.78    0.50  <2e-16 ***
## s(Timeliness)       9.00 4.11    0.56  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fitted_nb <- predict(model_nb_final, type = 'response')
```

```r
# PLOTTING STUFF - ERROR WHILE COMPILING - WILL FIX LATER
# par(mfrow = c(1,2))
# plot.map(log(fitted_nb) - log(TBdata$Population))
# plot.map(log(TBdata$TB) - log(TBdata$Population))

#temporal model
par(mfrow = c(2,2), pch = 20)
model_nb_time <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous, by = Year) + s(Urbanisation
summary(model_nb)
```

```
##
## Family: Negative Binomial(6.146)
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous) + s(Illiteracy) +
##     s(Urbanisation) + s(Density) + s(Poverty) + s(Poor_Sanitation) +
##     s(Unemployment) + s(Timeliness)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.42871    0.01094  -770.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                      edf Ref.df  Chi.sq  p-value
## s(Indigenous)      1.489  1.795  20.396 2.92e-05 ***
## s(Illiteracy)      1.008  1.017   0.246  0.63129
## s(Urbanisation)    6.634  7.773  24.089  0.00148 **
## s(Density)         4.579  5.672 132.693  < 2e-16 ***
## s(Poverty)         5.733  6.911  17.934  0.01516 *
## s(Poor_Sanitation) 6.123  7.297  73.103  < 2e-16 ***
## s(Unemployment)    5.798  7.000  62.050  < 2e-16 ***
## s(Timeliness)      4.101  5.097  64.474  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.861   Deviance explained = 43.9%
## -REML = 7237.2  Scale est. = 1         n = 1671
```

```r
model_nb$aic
```

```
## [1] 14391.19
```

```r
#spatial model
model_nb_space <- gam(formula = TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) + s(Dens
summary(model_nb_space)
```

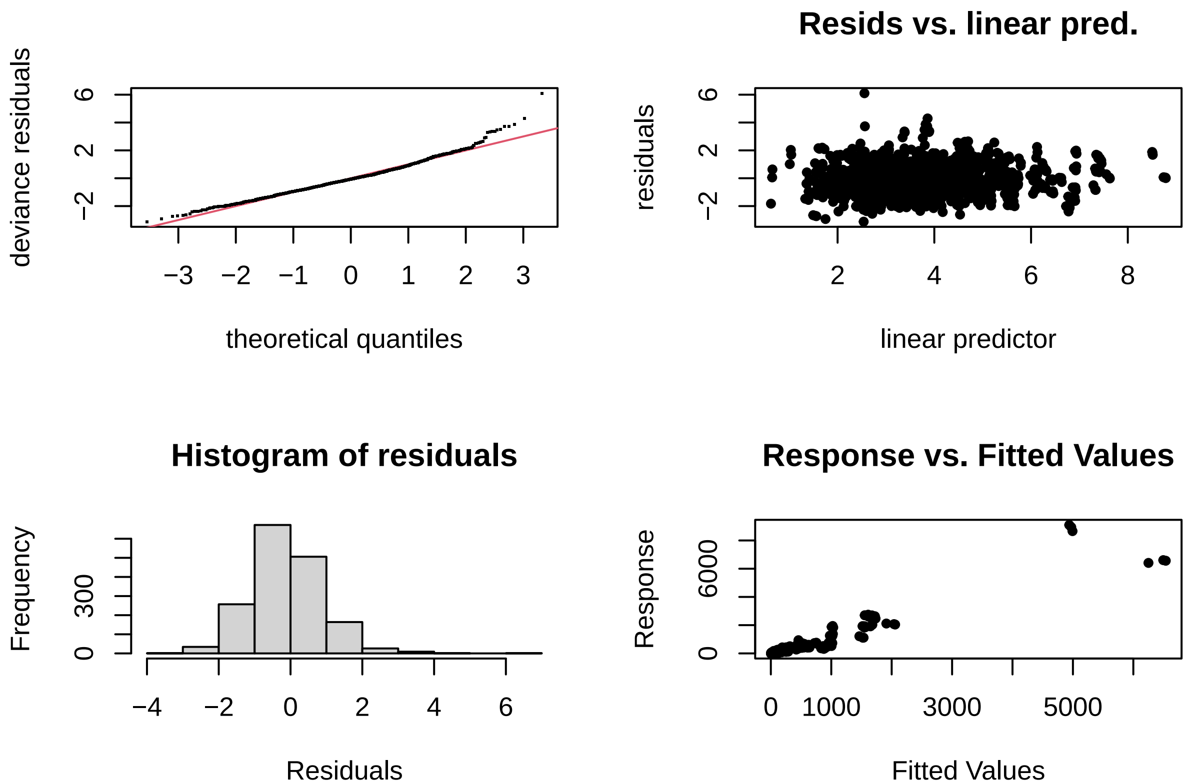
```
##
## Family: Negative Binomial(7.714)
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
```

```
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness) + te(lon, lat)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -8.443      0.010  -844.2   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                       edf Ref.df  Chi.sq  p-value
## s(Indigenous)       1.007  1.014  15.200 0.000101 ***
## s(Urbanisation)     4.458  5.567  22.483 0.000586 ***
## s(Density)          3.932  4.952  43.834  < 2e-16 ***
## s(Poverty)          2.131  2.721   8.551 0.027183 *
## s(Poor_Sanitation)  6.575  7.701  68.743  < 2e-16 ***
## s(Unemployment)     4.722  5.872  94.525  < 2e-16 ***
## s(Timeliness)       4.305  5.341  80.731  < 2e-16 ***
## te(lon,lat)        16.788 18.494 373.737  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.866   Deviance explained = 53.6%
## -REML = 7096.7  Scale est. = 1         n = 1671
```

```
model_nb_space$aic
```

```
## [1] 14094.26
```

```
gam.check(model_nb_space)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.002328308,0.0009125492]
## (score 7096.687 & scale 1).
## Hessian positive definite, eigenvalue range [0.002315995,532.0543].
## Model rank =  88 / 88
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                        k'   edf k-index p-value
## s(Indigenous)        9.00  1.01    0.52  <2e-16 ***
## s(Urbanisation)      9.00  4.46    0.53  <2e-16 ***
## s(Density)           9.00  3.93    0.52  <2e-16 ***
## s(Poverty)           9.00  2.13    0.52  <2e-16 ***
## s(Poor_Sanitation)  9.00  6.58    0.52  <2e-16 ***
## s(Unemployment)      9.00  4.72    0.53  <2e-16 ***
## s(Timeliness)        9.00  4.31    0.59  <2e-16 ***
## te(lon,lat)         24.00 16.79    0.48  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova.gam(model_nb_space, model_nb_final, test = 'LRT')
```

```
## Analysis of Deviance Table
##
```

```
## Model 1: TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness) + te(lon, lat)
## Model 2: TB ~ offset(log(Population)) + s(Indigenous) + s(Urbanisation) +
##     s(Density) + s(Poverty) + s(Poor_Sanitation) + s(Unemployment) +
##     s(Timeliness)
##   Resid. Df Resid. Dev      Df Deviance  Pr(>Chi)
## 1    1610.6      14004
## 2    1621.3      14319 -10.744  -314.19 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

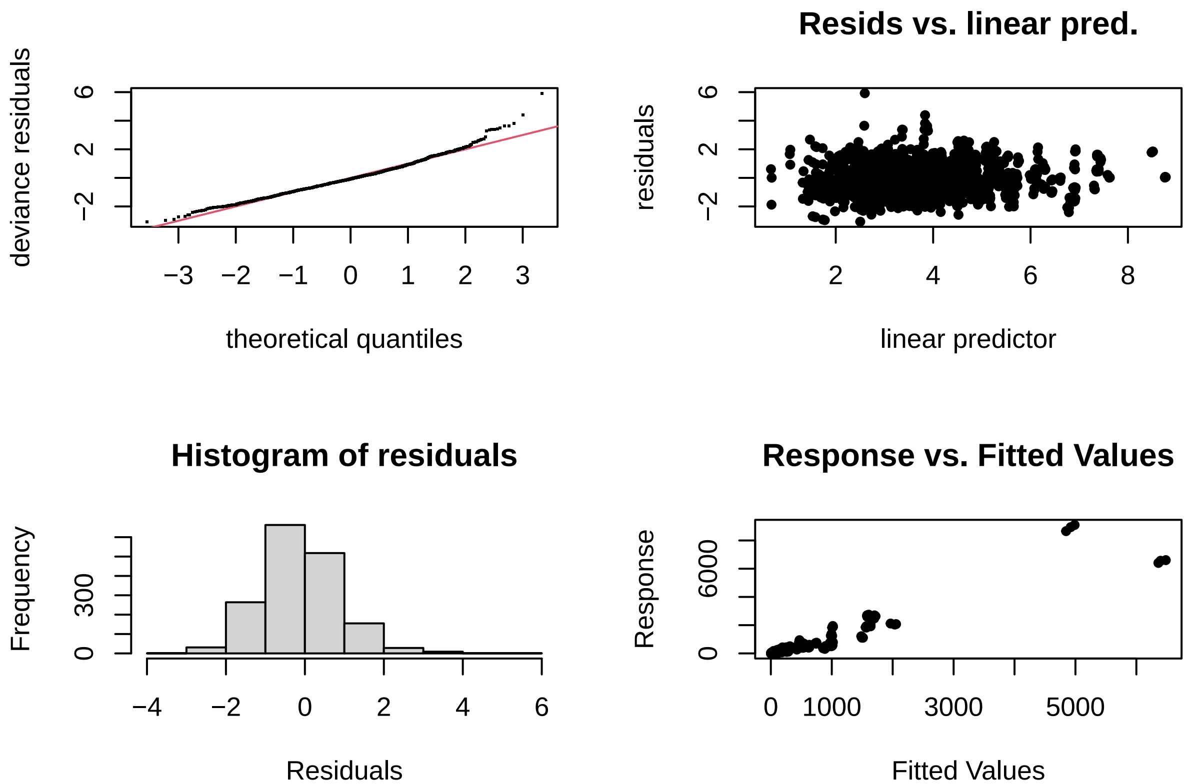
```r
#spatio-temporal model
model_nb_time_and_space <- gam(formula = TB ~ offset(log(Population)) + s(Urbanisation, by = Year) + s(I
summary(model_nb_time_and_space)
```

```
##
## Family: Negative Binomial(7.657)
## Link function: log
##
## Formula:
## TB ~ offset(log(Population)) + s(Urbanisation, by = Year) + s(Density,
##     by = Year) + s(Poverty, by = Year) + s(Poor_Sanitation, by = Year) +
##     s(Timeliness, by = Year) + s(Unemployment, by = Year) + te(lon,
##     lat, by = Year)
##
## Parametric coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)    31.80      24.43   1.302    0.193
##
## Approximate significance of smooth terms:
##                            edf Ref.df   Chi.sq p-value
## s(Urbanisation):Year     3.961  4.994    17.54 0.00323 **
## s(Density):Year          3.864  4.874    53.99 < 2e-16 ***
## s(Poverty):Year          2.796  3.548     7.42 0.06673 .
## s(Poor_Sanitation):Year  5.932  7.094    65.26 < 2e-16 ***
## s(Timeliness):Year       4.413  5.465 82638.12 < 2e-16 ***
## s(Unemployment):Year     5.805  6.963    96.46 < 2e-16 ***
## te(lon,lat):Year        17.115 18.790   384.46 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 80/86
## R-sq.(adj) =  0.863   Deviance explained = 53.3%
## -REML = 7165.9  Scale est. = 1         n = 1671
```

```r
model_nb_time_and_space$aic
```

```
## [1] 14106.85
```

```r
gam.check(model_nb_time_and_space)
```




```
##
## Method: REML   Optimizer: outer newton
## full convergence after 8 iterations.
## Gradient range [-0.0003475393,0.0002157822]
## (score 7165.925 & scale 1).
## Hessian positive definite, eigenvalue range [0.3757953,531.9567].
## Model rank =  80 / 86
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                              k'   edf k-index p-value
## s(Urbanisation):Year     10.00  3.96    0.52  <2e-16 ***
## s(Density):Year          10.00  3.86    0.52  <2e-16 ***
## s(Poverty):Year          10.00  2.80    0.53  <2e-16 ***
## s(Poor_Sanitation):Year 10.00  5.93    0.52  <2e-16 ***
## s(Timeliness):Year       10.00  4.41    0.59  <2e-16 ***
## s(Unemployment):Year     10.00  5.80    0.52  <2e-16 ***
## te(lon,lat):Year         25.00 17.11    0.48  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.