Effects of urban green infrastructure on supply provision of ecosystem service: Temperature reduction

Irina Lerner

30/01/24

Contents

Data **1**

Exploratory Analysis **2**

Response 2

Add Microclimatic region 2

Add Proportion of vegetation 6

Water bodies 9

Spatial considerations 10

parei **13**

Residuals 14

Refinement **18**

QUESTIONS **19**

CLEAR AIM: TO SHOW THE EFFECT OF VEGETATION CONFIGURATION IN DT.

Vegetation configuration is represented here in some different forms. - As building volume and height and vegetation volume and height for each sample - As SVI (sharing proportion), SUVI (Urban proportion) and SGVI (green propotion) and NA where data is invalid. We have Mean EVI as well to help us qualify vegetation.

I'm super unsure what is the best road to take. Every model seems to have downsides.

Data

The data set is a 100m^2 grid over the city of São Paulo. The temperature control variable is represented by DT, the difference among the cell temperature and the temperature of the local climatic zone.

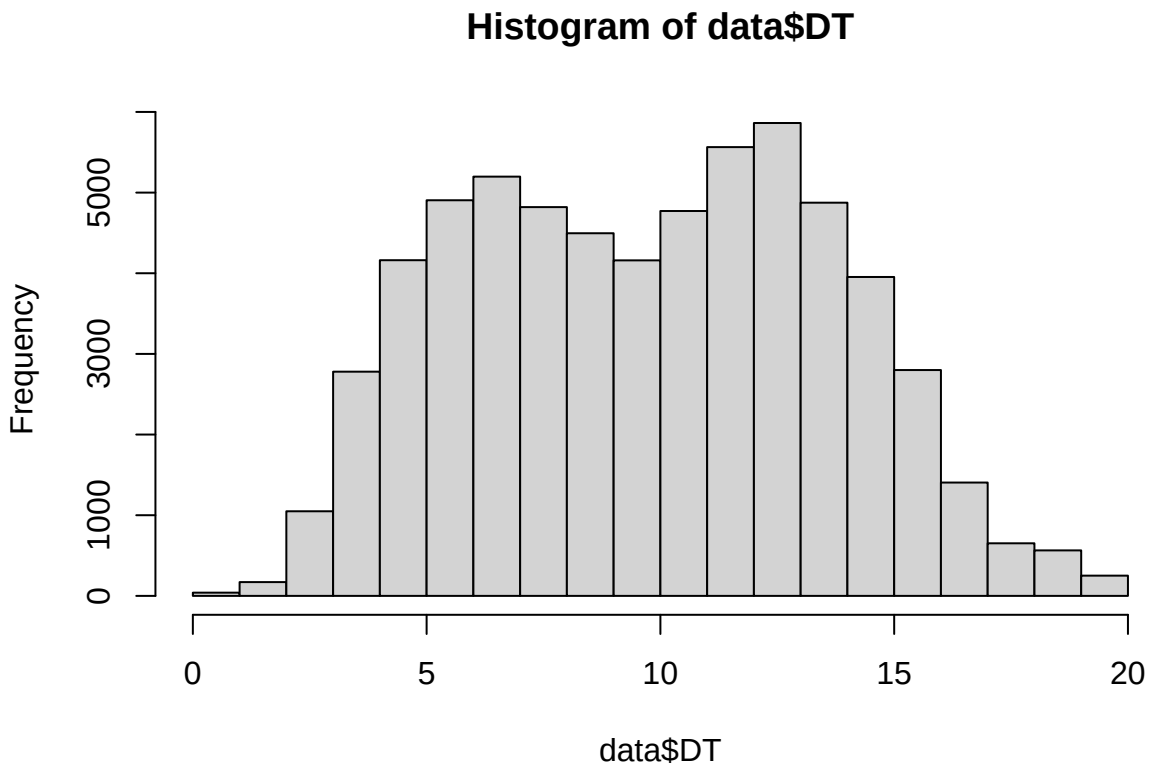
Proportion of vegetation is prop_veg and EVI for the region is EVI_mean (since EVI map is a 30m^2 grid, there is ~ 9 EVI cells in a 100m^2 grid).

Exploratory Analysis

Response

We begin exploring the response data.

```
hist(data$DT)
```



We can already see its not normally distributed. Let's uncover what lies behind this distribuition.

Add Microclimatic region

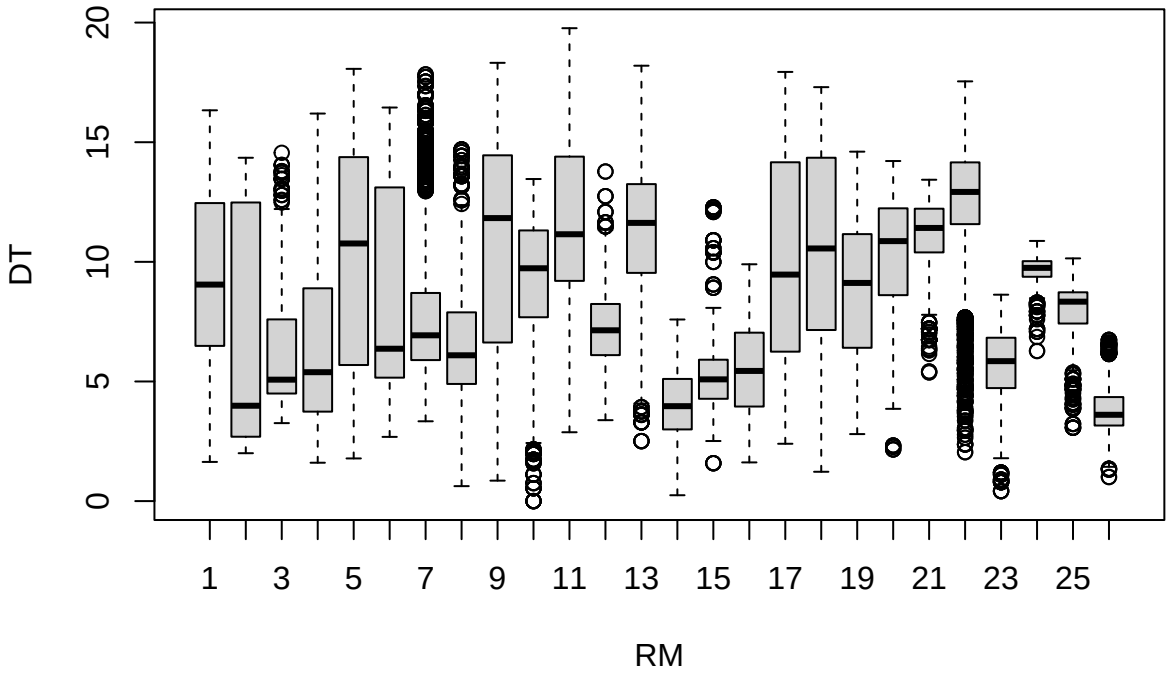
```
model <- lm(DT ~ RM, data)
pAIC <- analyse_linear_model(data, model)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 346978.7
```

```
anova(model)
```

```
## Analysis of Variance Table
##
## Response: DT
##               Df Sum Sq Mean Sq F value    Pr(>F)
## RM             1   4813  4812.8   318.5 < 2.2e-16 ***
## Residuals 62479 944105    15.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
boxplot(DT ~ RM, data)
```



Anova shows significant influence of the micro climatic region in each group. So either we add RM as a random factor or we normalize by RM. Lets try it all

Analyzing the effect of RM only

```
basic <- lm(DT ~ 1, data = data)
model_fixed <- lm(DT ~ RM, data = data)

library(lme4)
```

```
## Warning: package 'lme4' was built under R version 4.3.3
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.3.3
```

```
##
## Attaching package: 'lme4'

## The following object is masked from 'package:raster':
##
##     getData
```

```
model_random <- lmer(DT ~ 1 + (1 | RM), data = data, REML = F)

data$DT_norm <- with(data, DT - ave(DT, RM, FUN = mean))
model_norm <- lm(DT_norm ~ 1, data = data)

AIC(basic)
```

```
## [1] 347294.4
```

```
AIC(model_fixed)
```

```
## [1] 346978.7
```

```
AIC(model_norm)
```

```
## [1] 317066.9
```

```
AIC(model_random)
```

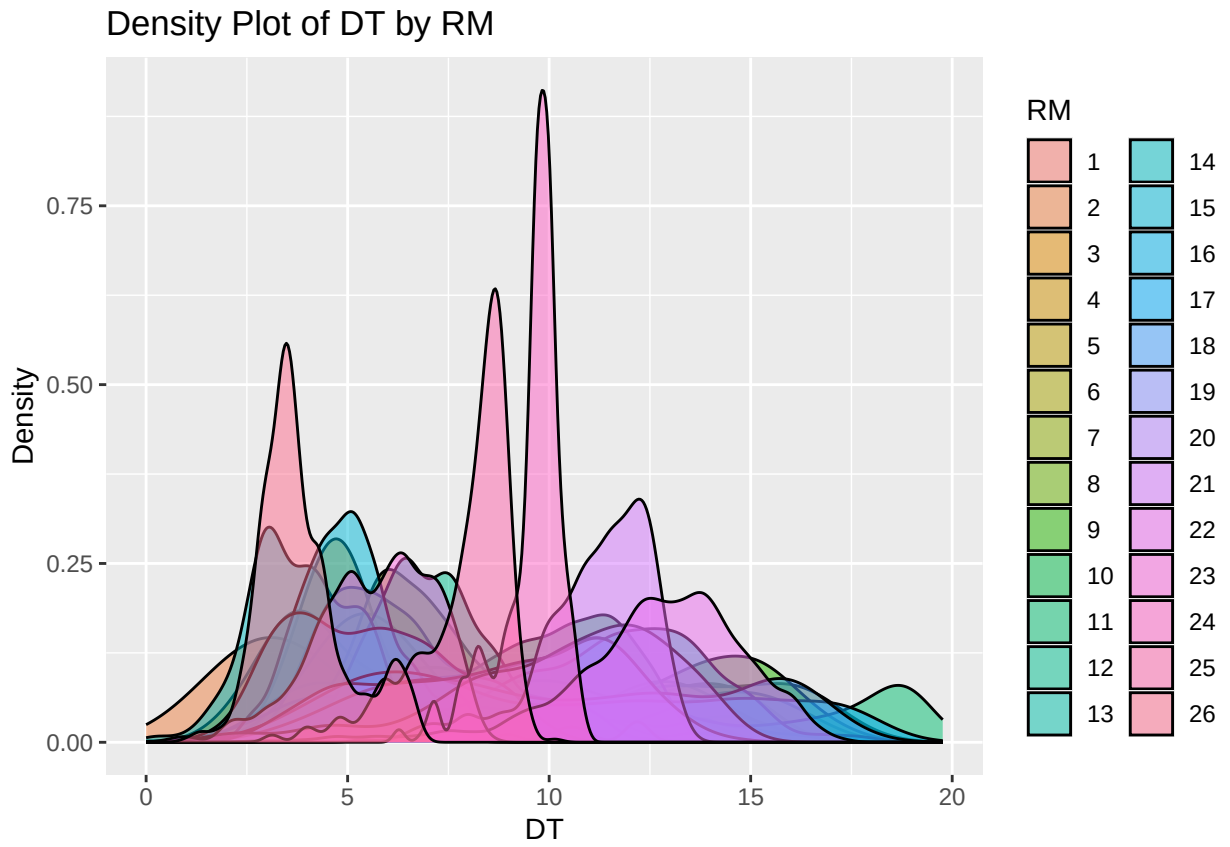
```
## [1] 317270.7
```

```
pAIC<- analyse_linear_model(data, model_norm)
```

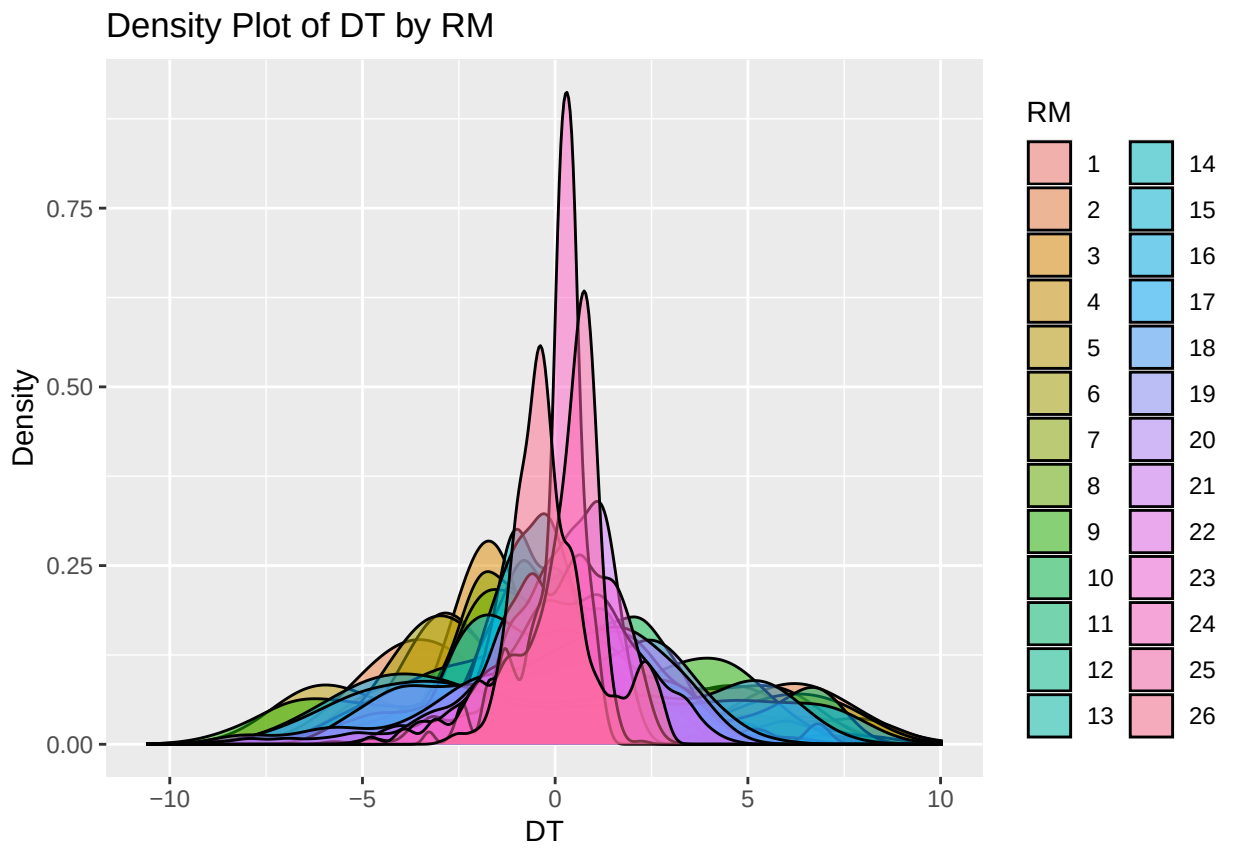
```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the left"
## [1] "model improved!"
## [1] "model stats"
## [1] 317066.9
```

Looking into each group's distribution, we can see that there is a lot of differences

```
ggplot(data, aes(x = DT, fill = factor(RM))) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plot of DT by RM", x = "DT", y = "Density", fill = "RM")
```




```
ggplot(data, aes(x = DT_norm, fill = factor(RM))) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plot of DT by RM", x = "DT", y = "Density", fill = "RM")
```



This looks way better.

Add Proportion of vegetation

Turning to a predictor that we know from literature to have an effect in DT: proportion of vegetation (prop_veg).

```
basic <- lm(DT ~ prop_veg, data = data)
model_fixed <- lm(DT ~ RM + prop_veg, data = data)
# Make sure to load the lme4 package for lmer()
library(lme4)
# Linear mixed-effects model with RM as a random effect
model_random <- lmer(DT ~ prop_veg + (1 | RM), data = data, REML = F)

# Normalize DT by RM
data$DT_norm <- with(data, DT - ave(DT, RM, FUN = mean))
# Linear model with normalized DT
model_norm <- lm(DT_norm ~ prop_veg, data = data)

AIC(basic)
```

```
## [1] 330271.3
```

```
AIC(model_fixed)
```

```
## [1] 328715.2
```

```
AIC(model_norm)
```

```
## [1] 306644.6
```

```
AIC(model_random)
```

```
## [1] 298048
```

```
pAIC<- analyse_linear_model(data, model_random)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 298048
```

This analysis suggests that when considering the effect of proportion of vegetation, the model with RM as a random effect seems like the best model. Residuals are still not normally distributed.

Let's dig in a little deeper into micro climatic regions. We are going to do one regression for each region and compare the effects

```
library(spdep)  # Ensure you have this package loaded for Moran's I

# Unique groups in RM
unique_rms <- unique(data$RM)
r_squared_data <- data.frame(RM = character(), R2 = numeric(), Moran_I = numeric(), Slope = numeric(),

# Loop through each RM to fit model, calculate R2, Moran's I, and Slope
for (rm in unique_rms) {
    # Subset data for each RM
    subset_data <- data[data$RM == rm, ]

    # Fit linear model
    model <- lm(DT ~ prop_veg, data = subset_data)

    # Calculate R2
    r_squared <- as.numeric(summary(model)$r.squared)

    nb <- poly2nb(subset_data)
    listw <- nb2listw(nb, style = "W", zero.policy = T)
    morans_i <- as.numeric(moran.test(residuals(model), listw)$estimate['Moran I statistic'])

    # Retrieve slope of the model
    slope <- as.numeric(coef(model)["prop_veg"])

    line <- data.frame(rm, r_squared, morans_i, slope)
    colnames(line) <- c("RM", "R2", "Moran_I", "Slope")

    # Append results to the data frame
```

```
    r_squared_data <- rbind(r_squared_data, line)
}

print(r_squared_data)
```

```
##    RM         R2   Moran_I     Slope
## 1  20 0.19836869 0.7808836  4.916886
## 2  19 0.73792596 0.6063779  7.613157
## 3  18 0.43789290 0.7780569  8.200983
## 4   3 0.75008445 0.6914133 10.608020
## 5  22 0.16744833 0.8090276  3.893330
## 6   9 0.45847781 0.7433770  8.722955
## 7  21 0.30369256 0.7326006  4.157040
## 8  10 0.36107459 0.7127647 -4.974562
## 9  13 0.04324572 0.7421143  2.298216
## 10 17 0.60226331 0.7605586  9.853998
## 11  4 0.81628488 0.6434878 10.657411
## 12 11 0.43698586 0.7557720  8.321770
## 13 25 0.09181569 0.7754422  2.404234
## 14 24 0.42084292 0.6745137  9.237084
## 15 26 0.02962084 0.8234420 -2.135465
## 16  5 0.59686280 0.5767622 10.734754
## 17 23 0.25649175 0.8005225 -2.423530
## 18  2 0.92872962 0.3891930 11.985471
## 19  6 0.48754815 0.7558810 10.780031
## 20 12 0.39584862 0.5836516  6.590120
## 21 16 0.60293355 0.6634085  8.256622
## 22  7 0.75290667 0.5751885  9.058040
## 23 14 0.26158655 0.7172142  4.871478
## 24  1 0.66599249 0.7124540  8.315871
## 25  8 0.64581751 0.6505069  7.723175
## 26 15 0.66082572 0.6051067  6.935591
```

We can see that models look a lot different for each sample, with different Moran I values

This table and the next image shows different predicted linear effects for different micro climatic regions. What lies behind this variation?

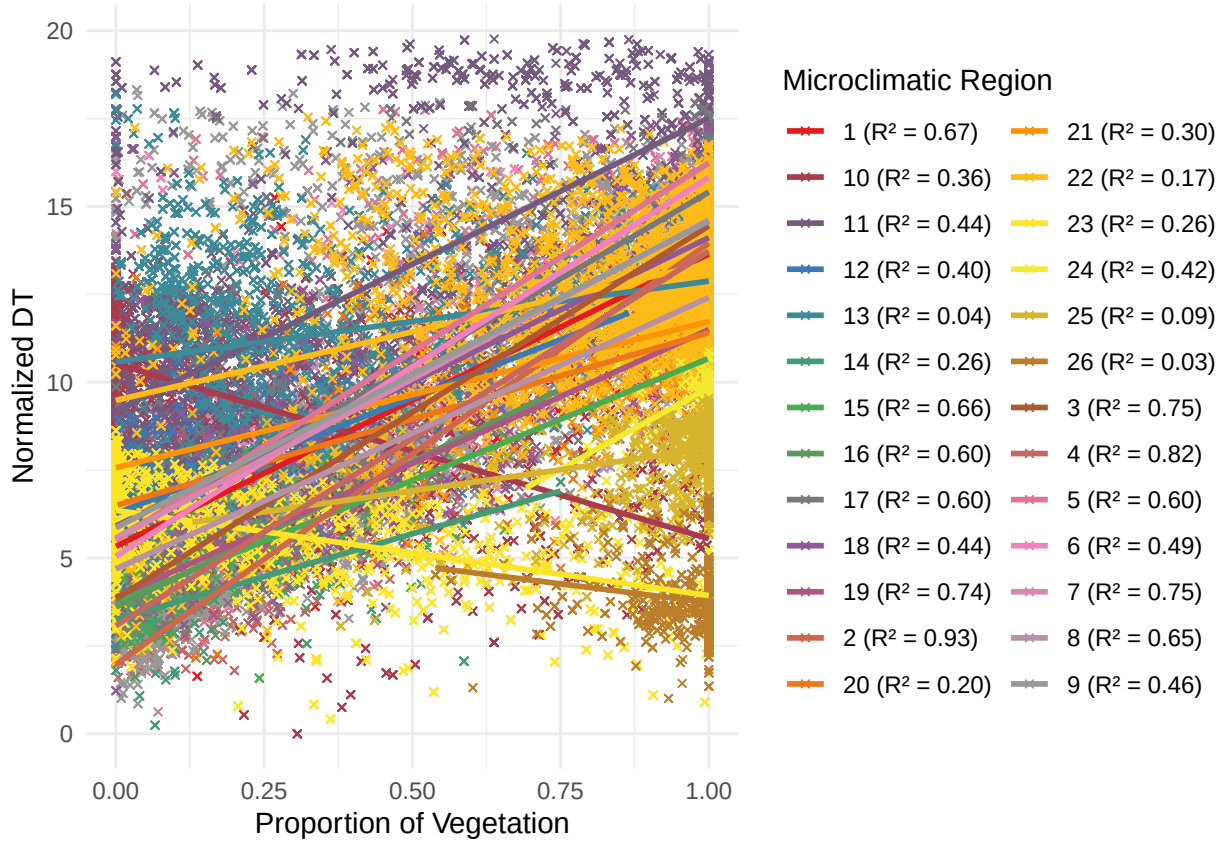
```
data_rm <- merge(data, r_squared_data, by = "RM", all.x = TRUE)

data_rm$RM_label <- with(data_rm, paste(RM, sprintf("(R² = %.2f)", (R2))))

colors <- colorRampPalette(RColorBrewer::brewer.pal(9, "Set1"))(26)

ggplot(data_rm, aes(x = prop_veg, y = DT, color = RM_label)) +
  geom_point(shape = 4, size = 1) +  # Points with custom shape and size
  geom_smooth(method = "lm", formula = y ~ x, se = FALSE) +  # Add linear model fit lines without stand
  scale_color_manual(values = colors) +
  theme_minimal() +  # Minimalist theme
  labs(
    x = "Proportion of Vegetation",
    y = "Normalized DT",
    color = "Microclimatic Region"  # Legend title
```

```
  ) +
  theme(legend.position = "right")  # Position the legend on the right
```



From this analysis, linear relationship with prop_veg is clear across most of the microclimatic regions, except those near water. This analysis help us narrow down data for analysis that fit our assumptions and refine our models.

Water bodies

If we also add the presence of water as a predictor

```
model_random_water <- lmer(DT ~ prop_veg + prop_water + (1 | RM), data = data, REML = F)
model_residual_water <- lm(residuals(model_random) ~ data$prop_water)

AIC(basic)
```

```
## [1] 330271.3
```

```
AIC(model_random)
```

```
## [1] 298048
```

```
AIC(model_random_water)
```

```
## [1] 290555.9
```

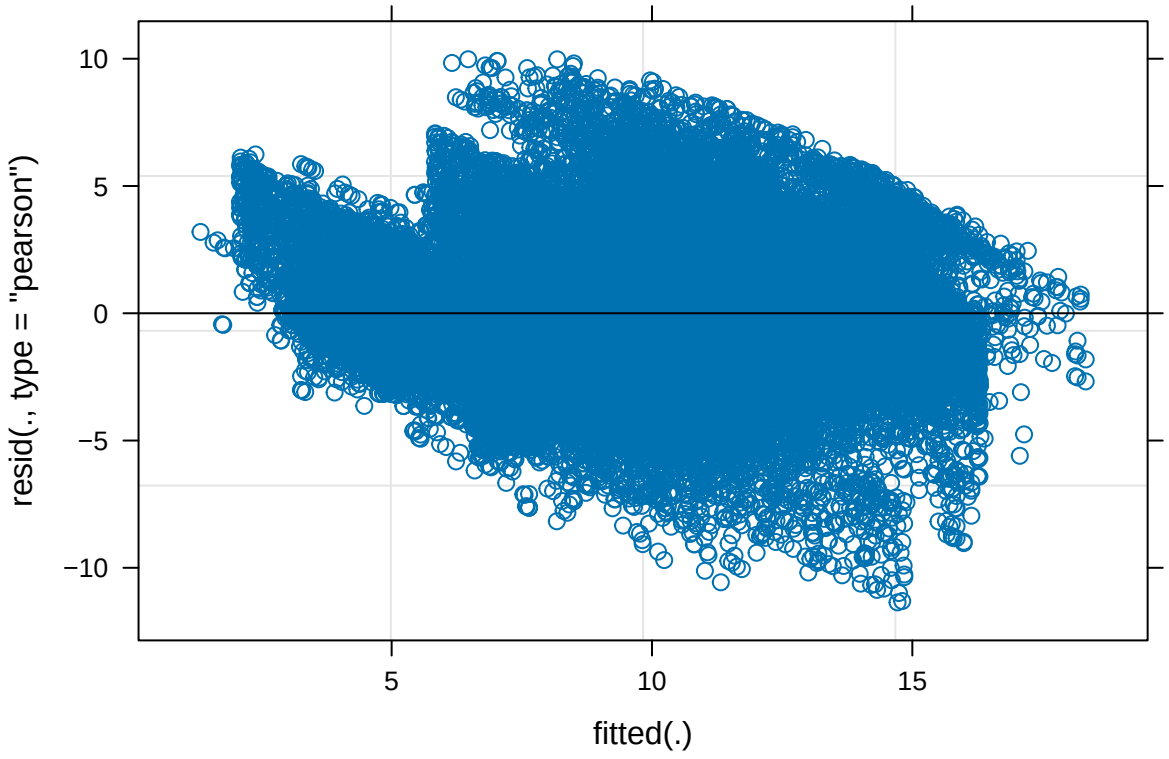
```
AIC(model_residual_water)
```

```
## [1] 292878.4
```

```
pAIC<- analyse_linear_model(data, model_random_water, pAIC)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 290555.9
```

```
plot(model_random_water)
```



There is still non-normallity of residuals and a pattern seen. But this is our best model so far.

Spatial considerations

Another factor that might be really significant is the spatial correlation, since heat has a spatial dispersion effect like sinks and sources.

```r
# For example, defining neighbors based on contiguity (sharing a boundary):
neighbors <- poly2nb(data)
weights <- nb2listw(neighbors, style="W", zero.policy=TRUE)
moran.test(residuals(model_random_water), weights)
```

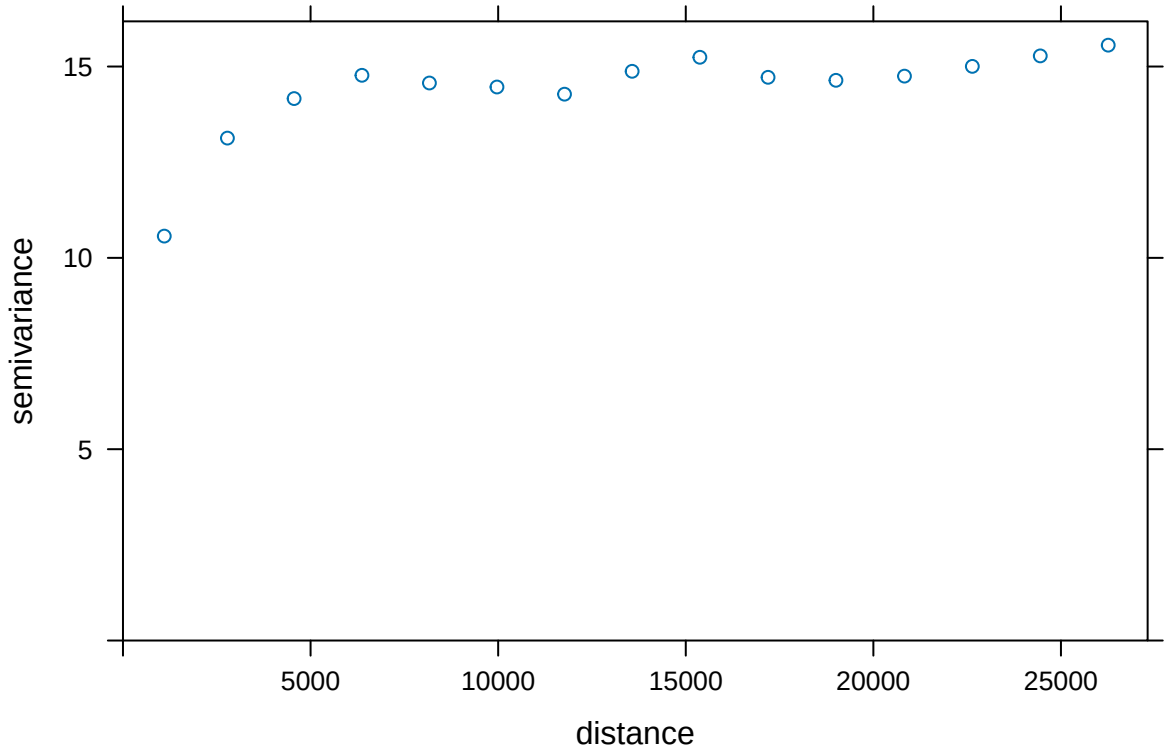
```
##
##  Moran I test under randomisation
##
## data:  residuals(model_random_water)
## weights: weights
## n reduced by no-neighbour observations
##
## Moran I statistic standard deviate = 472.61, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      6.810925e-01     -1.607820e-05      2.076931e-06
```

Moran I test shows a lot of spatial correlation not accounted for in the models.

To understand space, we first create a variogram that takes a lifetime to run so it's not included here.

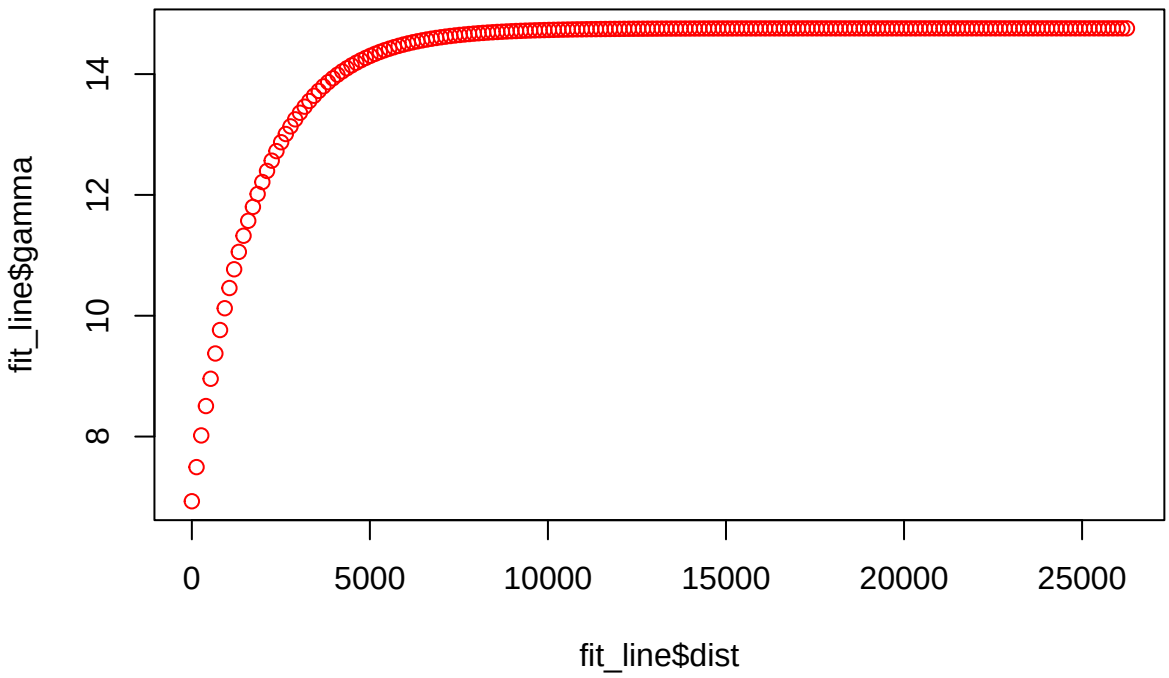
```r
library(sp)
library(gstat)

# Create variogram
variogram <- variogram(DT ~ 1, data)
plot(variogram)
```




```r
# Fit a variogram model
fit_variogram <- fit.variogram(variogram, model = vgm(1, "Exp", 10000, 1))

# Generate a line for the fitted variogram model over a sequence of distances
fit_line <- variogramLine(fit_variogram, maxdist = max(variogram$dist))
# Add the line for the fitted model to the plot
plot(fit_line$dist, fit_line$gamma, col = "red")
```



The variogram shows a possible exponential effect of distance.

From now on, to build a model, we need to include: 1 - Micro climatic region as a random factor or normalized DT 2 - Spatial relationships preferentially with exponential effect.

Not all models do that, and that is the tricky part.

parei

PAREI AQUI TO TESTANDO OS MODELOS ESPACIAIS. SEPA VOLTAR PRO FITME.

I've tried all models with spatial relationships and GAMMs seem to be the best for us right now. Since prop veg is linear but RM cam be added as a smooth random factor. Lets reproduce and compare results

```
#AIC(model_random_water)
## adding int and slope to model
#gamm_copy <- gamm(DT ~ prop_veg + prop_water,
#                        random = list(RM = ~ 1), data = data, method = "ML")
#AIC(gamm_copy$lme)

#gamm_space_smooth <- gamm(DT ~ prop_veg + prop_water + s(centx, centy, bs = "gp", m = 1),
#                        random = list(RM = ~ 1), data = data, method = "ML")

# Fit a GAMM with a Gaussian process term
#gamm_space_cor <- gamm(DT ~ prop_veg + prop_water,
#                       random = list(RM = ~ 1),
```

```
#                       data = data,
#                       method = "ML",
#                       correlation = corExp(form = ~ centx + centy))

#moran.test(residuals(gamm_space_smooth$gam), weights)
#moran.test(residuals(gamm_space_cor$gam), weights)
```

The AIC is the same. Now lets include space as covariance. this is not dealing with the space properly

```
#library(nlme)

# Fit a linear mixed-effects model with spatial correlation structure
#lme_spatial <- lme(DT ~ prop_veg + prop_water,
#                     random = ~ 1 | RM,
#                     correlation = corExp(form = ~ centx + centy),
#                     data = data,
#                     method = "ML")

#moran.test(lme_spatial)
#AIC(lme_spatial)
```

Residuals

The deal is that relationship is pretty linear but the residuals are never normally distribuited. One analytical resource we have is to analyse the residuals from a linear model against a group of variables.

```
resid_basic <- lm(residuals(model_random_water) ~ 1, data)
pAIC <- analyse_linear_model(data, resid_basic)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 290341.4
```

```
resid_BV <- lm(residuals(model_random_water) ~ BV, data)
pAIC <- analyse_linear_model(data, resid_BV, pAIC)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 283592.9
```

```
resid_VVBV <- lm(residuals(model_random_water) ~ BV + VV, data)
pAIC <- analyse_linear_model(data, resid_VVBV, pAIC)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 283268.5
```

```
resid_inter <- lm(residuals(model_random_water) ~ BV * VV, data)
pAIC <- analyse_linear_model(data, resid_inter, pAIC)
```

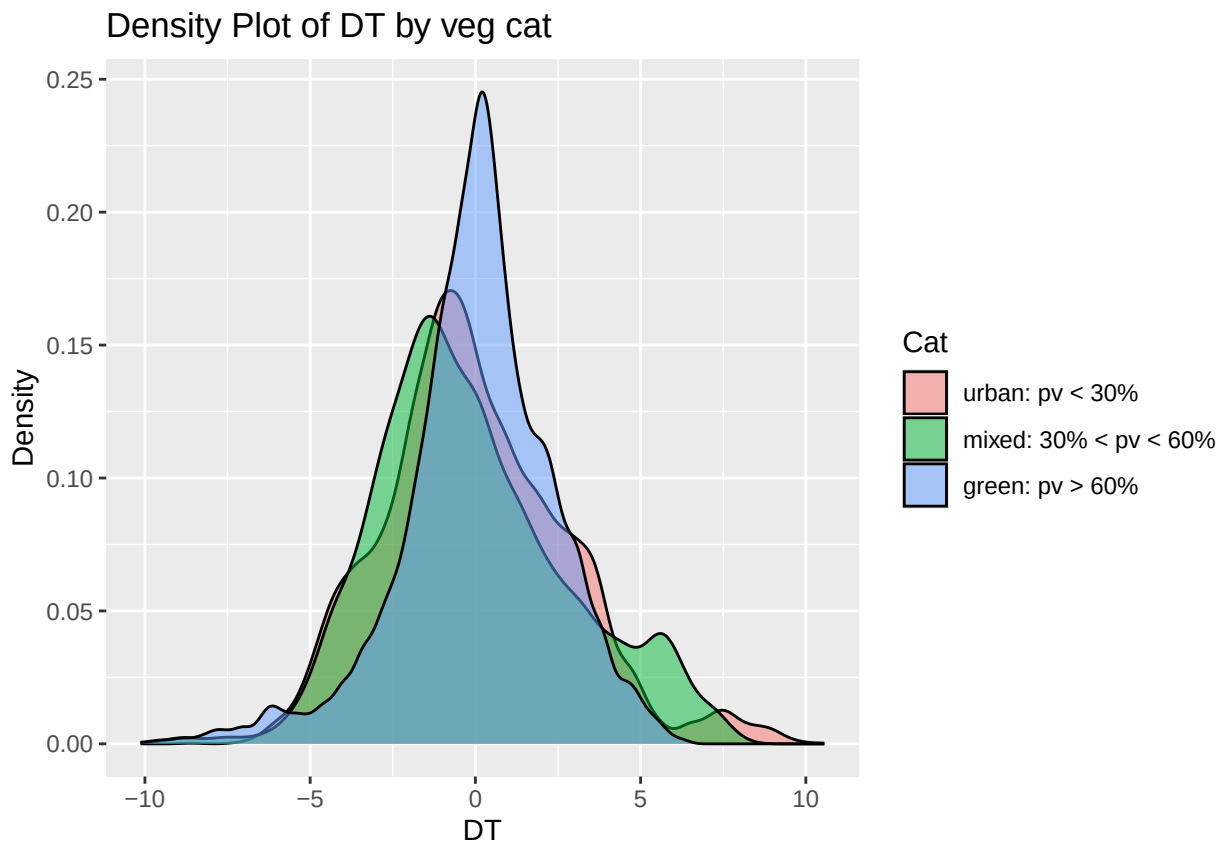
```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model improved!"
## [1] "model stats"
## [1] 283260.9
```

```
resid_EVI <- lm(residuals(model_random_water) ~ EVI_mean, data)
pAIC <- analyse_linear_model(data, resid_inter, pAIC)
```

```
## [1] "Residuals not normally distributed:"
## [1] "They're skewed to the right"
## [1] "model is worse than before"
```

Lets see if the distribution varies in prop_veg categories,

```
ggplot(data, aes(x = residuals(model_random), fill = factor(Category))) +
  geom_density(alpha = 0.5) +
  labs(title = "Density Plot of DT by veg cat", x = "DT", y = "Density", fill = "Cat")
```



Seems like the variance for residuals is higher in mixed areas.

I can see some scenarios for this to be happening.

- the more homogeneous the area, in a bigger scale, the less variance in temperature we will see. There is less possible configurations for the landscape. The stronger the spatial effect as well
- land sharing plays a role
- land sparing plays a role
- spatial effects are not accounted for
- built volume and height plays a role.

specially in areas considered green. What is happening here?

A proxy for vegetation amount is the EVI. EVI is a bit more explanatory than proportion of vegetation. It considers the mean color of the surface, considering green and grey.

"The Enhanced Vegetation Index (EVI) is a satellite-derived index designed to optimize the vegetation signal with improved sensitivity in high biomass regions and improved vegetation monitoring through a de-coupling of the canopy background signal and a reduction in atmosphere influences.

EVI is an improvement over the more common Normalized Difference Vegetation Index (NDVI), which can be saturated in high biomass areas. This saturation can make NDVI less sensitive to differences in dense vegetation. Additionally, NDVI can be influenced by soil background and atmospheric conditions, which can obscure the true vegetation signal.

EVI better detects and quantifies:

- Vegetation vigor and productivity
- Canopy structural variations, including leaf density and type, as well as plant phenology
- Changes in vegetation cover, including monitoring of vegetation dynamics over time
- Stress and disturbances in plant growth Agricultural monitoring and forecasting"

We can see that variance is different in each category. Also, the model does improves from vegetation proportion. The model has a very low AIC showing a clear relationship, but residuals are not normally distributed. Differences in extremities may turn it a bit.

Adding other components make models worse, but they're not co linear as VIF shows. Summary indicates that each of these elements affects

Turning now to other families for our models, to generalized mixed models.

```
library(lme4)

# Assuming 'data' is your dataset
# 'DT' is your response variable (transformed if necessary)
# 'RM' is the random effect
# 'prop_veg', 'EVI_mean' are fixed effects

data_positive <- data[log(data$DT) > 0,]

glmm_model <- glmer(log(DT) ~ prop_veg + (1 | RM),
                    data = data_positive,
                    family = inverse.gaussian(link = "log"))
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, :
## Model failed to converge with max|grad| = 0.0154284 (tol = 0.002, component 1)
```

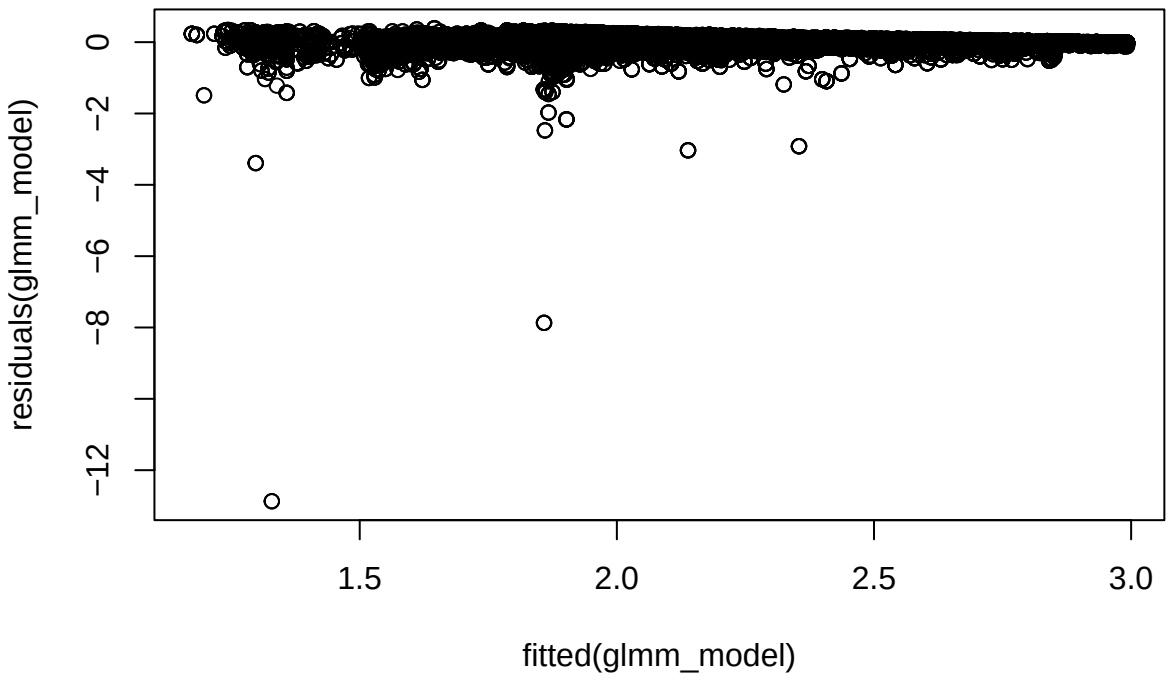
```r
# Check the model summary for coefficients and significance
summary(glmm_model)
```

```
## Generalized linear mixed model fit by maximum likelihood (Laplace
##   Approximation) [glmerMod]
##  Family: inverse.gaussian  ( log )
## Formula: log(DT) ~ prop_veg + (1 | RM)
##    Data: data_positive
##
##      AIC      BIC   logLik deviance df.resid
##  93002.2  93038.3 -46497.1  92994.2    62437
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -7.6664 -0.4344  0.0205  0.5411  4.3923
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  RM       (Intercept) 0.0009274 0.03045
##  Residual             0.0126868 0.11264
## Number of obs: 62441, groups:  RM, 26
##
## Fixed effects:
##             Estimate Std. Error t value Pr(>|z|)
## (Intercept) 0.501475   0.037887   13.24   <2e-16 ***
## prop_veg    0.316634   0.002475  127.96   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##          (Intr)
## prop_veg -0.032
## optimizer (Nelder_Mead) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.0154284 (tol = 0.002, component 1)
```

```r
AIC(glmm_model)
```

```
## [1] 93002.16
```

```r
# Plot residuals to check for patterns
plot(residuals(glmm_model) ~ fitted(glmm_model))
```



Refinement

To be able to do that, we need a modeling technique that can incorporate: - Random factors - Linear and Non-linear relationships - Spatial features

Let's turn our attention to GAMM models.

Fixed Effects s(prop_veg): This specifies a smooth term for prop_veg using a spline. This allows the model to capture non-linear relationships between the proportion of vegetation and DT (the response variable).

Random Effects (random = list(RM = ~ 1)): This includes random intercepts for RM, which accounts for variation in DT that is attributable to different levels of RM but not explained by the observed variables.

Spatial Correlation (correlation = corSpatial(form = ~ centx + centy, type = "exponential")): This specifies an exponential spatial correlation structure based on the coordinates centx and centy. It models the decay of correlation between observations as a function of distance, assuming that points closer together are more similar than points further apart.

Estimation Method (method = "REML"): REML (Restricted Maximum Likelihood) is used to estimate the model parameters. REML is often preferred over ML (Maximum Likelihood) when fitting models with random effects because it provides unbiased estimates of variance and covariance parameters.

```
# Model with random spatial effects in residuals
#gamm_model_random <- gamm(DT ~ s(prop_veg),
#                          random = list(RM = ~ 1),
#                          data = data,
#                          correlation = corSpatial(form = ~ centx + centy, type = "exponential"),
```

```
#                           method = "REML")
#AIC(gamm_model_random$lme)
#summary(gamm_model_random$gam)

#plot(resid(gamm_model_random$gam) ~ fitted(gamm_model_random$gam))
#moran.test(resid(gamm_model_random$gam), listw = weights)

#bam_model <- bam(DT ~ s(prop_veg)
#                    + s(centx, centy)
#                    + s(RM, bs = "re"),
#                    data = data_positive,
#                    family = Gamma(link = "log"),
#                    discrete = TRUE,  # This enables faster computation for large datasets
#                    nthreads = 4)  # threads for paralel computation

#summary(bam_model)
#neighbors <- poly2nb(data_positive)
#weights <- nb2listw(neighbors, style="W", zero.policy=TRUE)
#moran.test(resid(bam_model), weights)
```

QUESTIONS

For Artur: is RM already included in DT? Or only in DT_wait? o que ta acontecendo na RM 25?

For Doug: What does the EVI actually represents?

For Melina: Why would I use a NB dist for pop if I'm using density? Should we change?

(The amount of NA shows I have to reacess my data)