GEOG0125

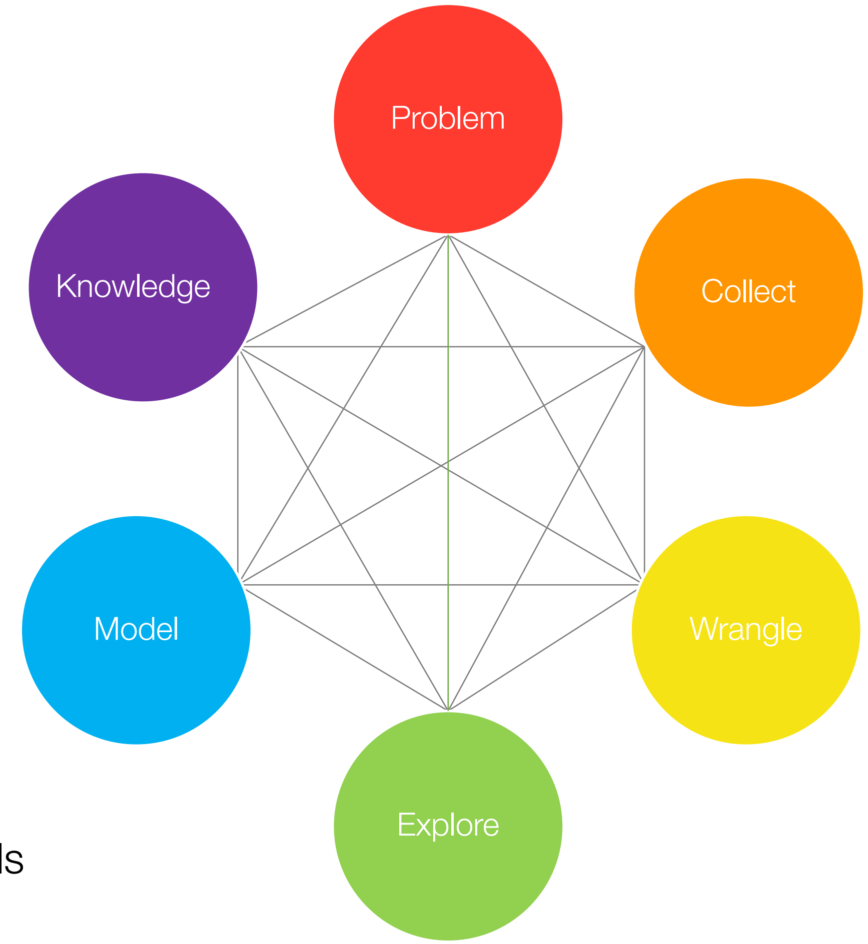
ADVANCED TOPICS IN SOCIAL AND GEOGRAPHIC DATA SCIENCE

INTRODUCTION TO BAYESIAN GENERALISED ADDITIVE MODELS (GAM)

Dr Anwar Musah (a.musah@ucl.ac.uk)
Lecturer in Social and Geographic Data Science
UCL Geography

Contents

- What are Generalised Additive Models (GAMs)?
  - Usage: For exploring non-linear relationships
  - Importance: Notable applications in assessing or generating a so called "dose-response curve"
  - Trades-off in Model Building: Linear vs. Machine Learning
- Model components of GAMs
  - Polynomials
  - Basis functions & Smoothing
- Example and interpretation
- Model Specification from a Bayesian Framework
- RStudio
  - The package for implementing GAMs is called Bayesian Regression Models in Stan (BRMS).
  - Uses the **brm()** and **brm::stancode()** (translates the RStudio code directly into Stan code – incredibly useful)



Remember in Week 2’s we covered…

Recap on Definition:

Generalized linear model (GLMs) is a flexible generalization of ordinary linear regression model, which allows the user to link some outcome *y*, to a link function $g(\eta)$, when that outcome is characterised by distribution that is from one the exponential families of distribution.

$$g(\eta) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \varepsilon$$

Exponential family are set of parametric (i.e., discrete or continuous) probability distributions. There are many… but the most common examples are:

- Normal
- Binomial
- Poisson
- Multinomial
- Negative binomial

Notes 1: There are a tonne of them, but you really don't have to worry about any of them. You only need to concern yourself with how this link function works!

We covered link functions $g(\eta)$

Here are the most frequent examples which you will certainly encounter

Distribution of dependent variable	Exponential Family (Distribution)	Link Function	Suitable Model
Continuous measures	Normal distribution	Identity (we've been using this all this while)	Linear regression
Binary measures (1 = "present" or 0 = "absent")	Bernoulli distribution	Logit	Logistic Regression
Binomial measure (or proportion)	Binomial distribution	Logit function on aggregated outcome for successful and failures	Logistic Regression
Counts or discrete measures	Poisson distribution	Log or ln	Poisson Regression

In terms of regression, there are several types of models, each with there own families depending on the type distribution for the dependent variable:

Notes 1: Recall that we only touched on the fact the outcomes can measures that are from a different distribution, but we never really touched on this matter and on these particular classes of regression models

Here is a board overview:

Distribution of dependent variable	Suitable Model
Continuous measures: e.g., average income in postcode (£); concentrations of ambient particular matter (PM2.5); Normalised Vegetative Difference Index (NDVI) etc.,	Linear regression
Binary measures (1 = "present" or 0 = "absent"): e.g., Person's voting for a candidate, Lung cancer risk, house infested with rodents etc.,	Logistic Regression
Binomial measure (or proportion): e.g., prevalence of houses in a postcode infested with rodents, percentage of people in a village infected with intestinal parasitic worms, prevalence of household on a street segment victimised by crime etc.,	Logistic Regression
Counts or discrete measures: e.g., number of reported burglaries on a street segment, number of riots in a county etc.,	Poisson Regression
Time-to-event binary measures: e.g., Lung cancer risk due to chronic exposure to environmental levels of indoor radon. Risk of landslide and time dependence of surface erosion etc.,	Survival Analysis with Cox regression

What are Generalised Additive Models (GAMs)?

Definition:

Generalized additive model (GAMs) is a **flexible generalization of any ordinary regression model**, as well as it is a **smoothing technique** which allows the user to model **non-linear relationships** between an outcome *y* with a set of other independent variables *x*.

The typical statistical formulation of GAM: $\boldsymbol{y_i = \alpha + f_1(x_{1,1}) + f_2(x_{1,2}) + \cdots + f_p(x_{i,p}) + \varepsilon}$

What are GAMs exactly (in plain English):

- With linear models, we have explored to date are the considered as the "go-to" models. We have seen many adaptations (i.e., non-spatial and spatial) on a variety of conditions (i.e., Gaussian, Binomial and Poisson) and data types (i.e., continuous, binary or counts/rates).
- GAMs are simply an adaptation of a Generalised Linear Model (GLM) that can deal with non-linear data while maintaining explainability.

Re-formulation of a GLM to GAM:

Equation 1 is the formulation of a GLM: $$\boldsymbol{y_i = \beta_0 + \beta_1 x_{1,1} + \beta_2 x_{1,2} + \cdots + \beta_p x_{i,p} + \varepsilon}$$

Note 1: We should be familiar with this model now!

Equation 2 is the formulation of a GAM: $$\boldsymbol{y_i = \alpha + f_1(\mathrm{x}_{1,1}) + f_2(\mathrm{x}_{1,2}) + \cdots + f_p(x_{i,p}) + \varepsilon}$$

What are GAMs exactly (in plain English):

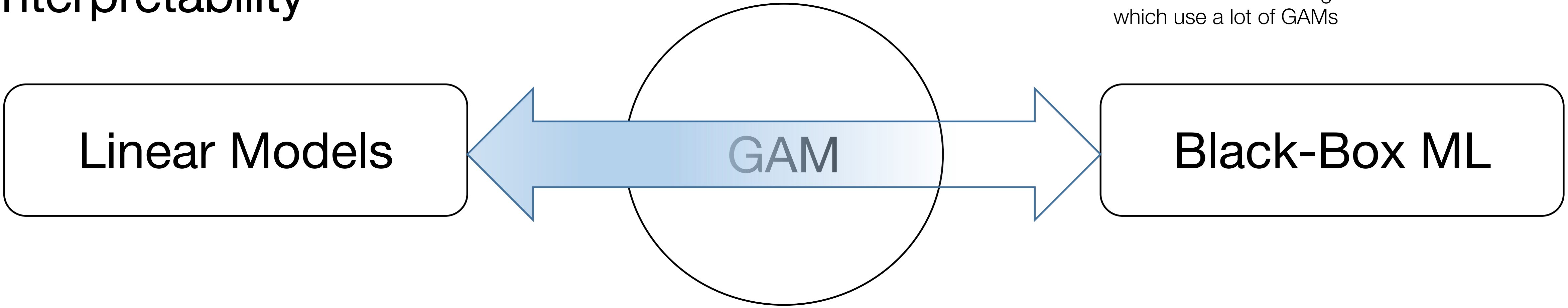
- GAMs are much more relaxed in their assumptions about linearity.
- The coefficients from a linear regression i.e., β_p are replaced with a **flexible function** i.e., f_p called a **Spline**, which are mathematical devices to enable the modelling of non-linear (or "wiggly") relationships between our outcome and independent variables.
- The sum of many splines hence forms a GAM, thus results in a highly flexible model (aka **pretzel-tier status**) which is still has some of the explainability of a generalised linear regression

Note 2: I will come to the maths in a second. As you will see, the flexible functions are something as simple as incorporating a quadratic, cubic and higher order functions into equation 2 as splines and making each variable a function.

Note 3: Interpretations are mostly through visual outputs

Trade-offs between Flexibility and Interpretability

Note 1: You will be taught ML neural networks which use a lot of GAMs



- Linear models are easy to interpret and to use for inference: It is easy to understand the meaning of their parameters. However, we often need to model more complex phenomena than can be represented by linear relationships.
- On the other hand, machine learning models, like boosted regression trees or neural networks, can be very good at making predictions of complex relationships. The problem is that they tend to need lots of data, are quite difficult to interpret, and one can rarely make inferences from the model results.
- GAMs offer a middle ground: they can be fit to complex, nonlinear relationships and make good predictions in these cases, but we are still able to do inferential statistics and understand and explain the underlying structure of our models and why they make predictions that they do.

Source: [GAMs in R by Noam Ross]


Zoology

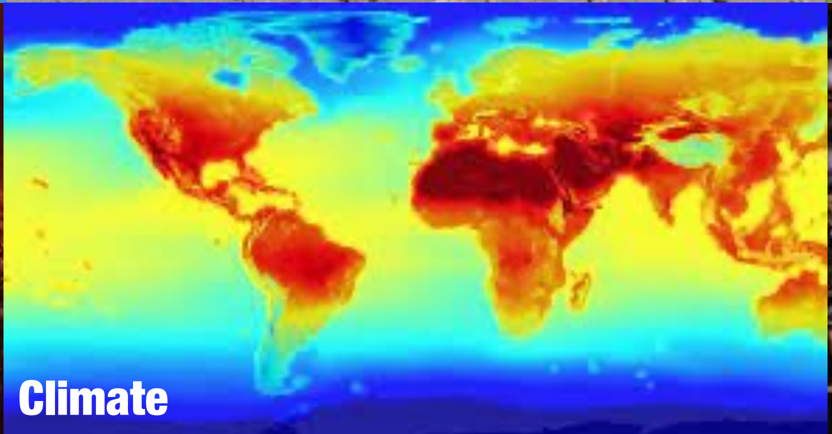
Food security

Environmental Criminology

Environmental & Spatial Epidemiology
Vector-borne disease

Palaeontogy and Archology

Landscape ecology

Natural Disaster Science

Climate

Humanitarian crisis

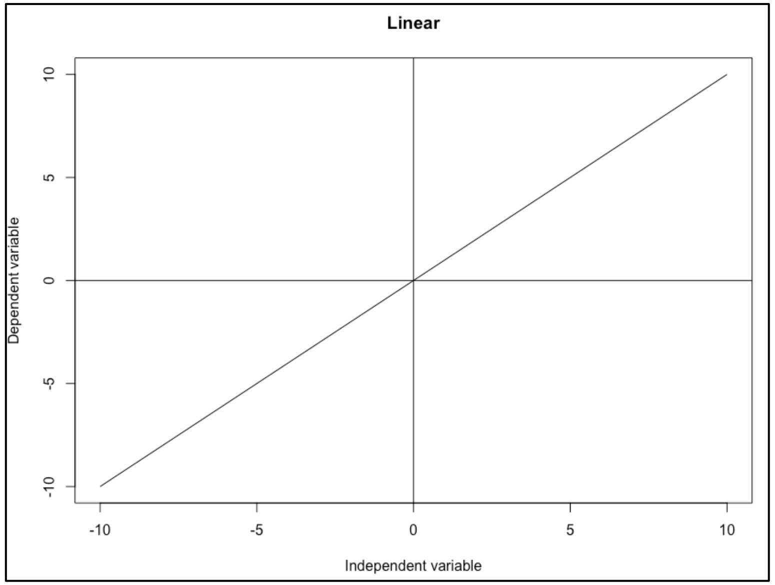
Model Components of a GAM

Maths 101: Polynomial functions [1]

- Polynomial functions is a mathematical device – when an independent variable is expressed with some kind power. Usually, it should be a power that is of an integer with a non-negative value.

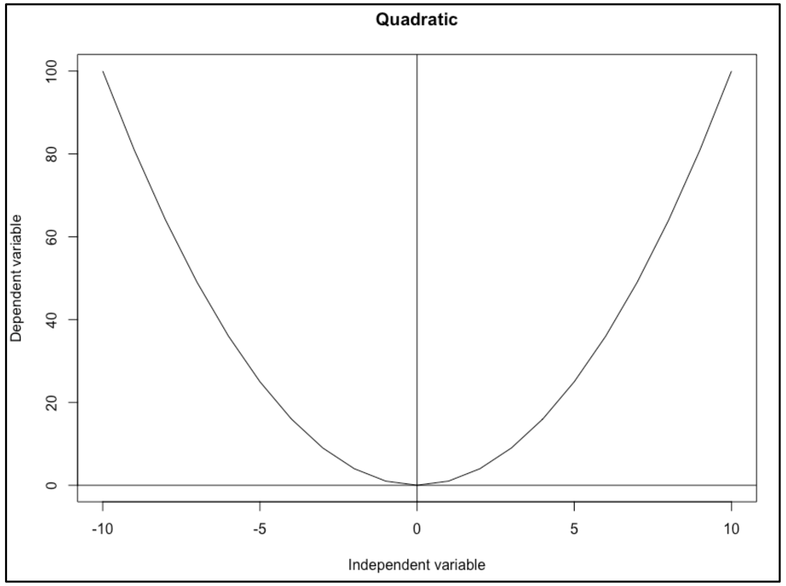
- Linear (1), Quadratic (2), Cubic (3) and polynomial functions with a higher degree (i.e. powers with 4 and onwards)

Linear function: $y = x$



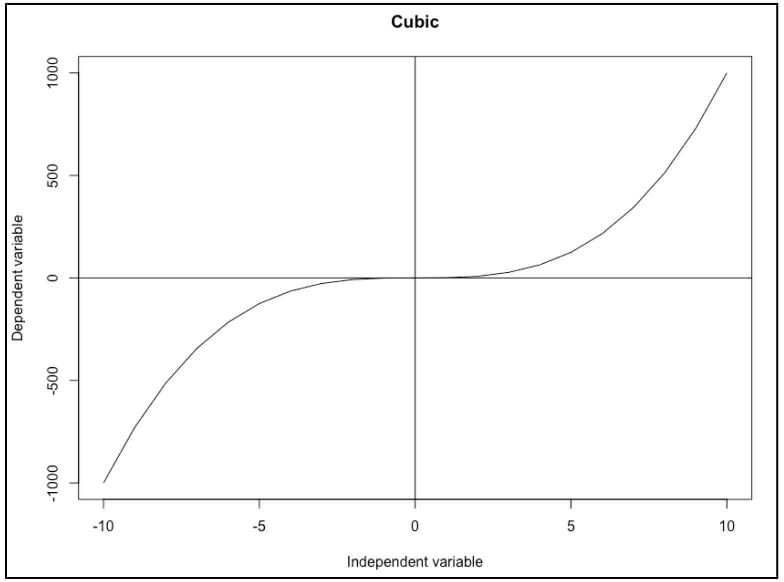
The effect of x on y is said to be **linear**

Quadratic function: $y = x^2$



The effect of x on y is said to be **quadratic** or **U-shaped**

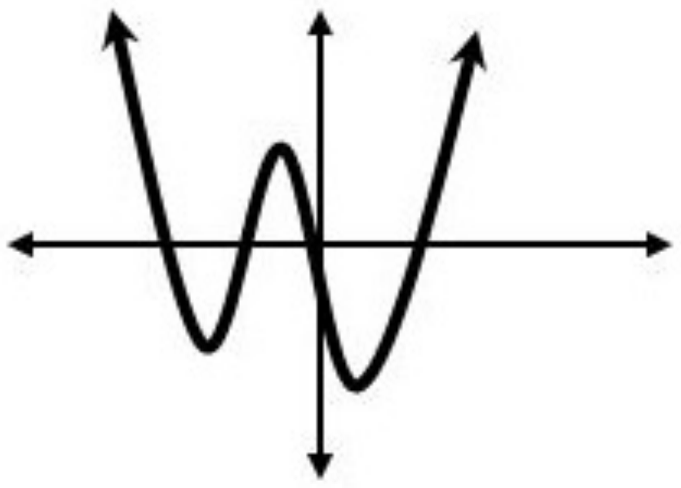
Cubic function: $y = x^3$



The effect of x on y is said to be **S-shaped that's not only inverted but rotated**

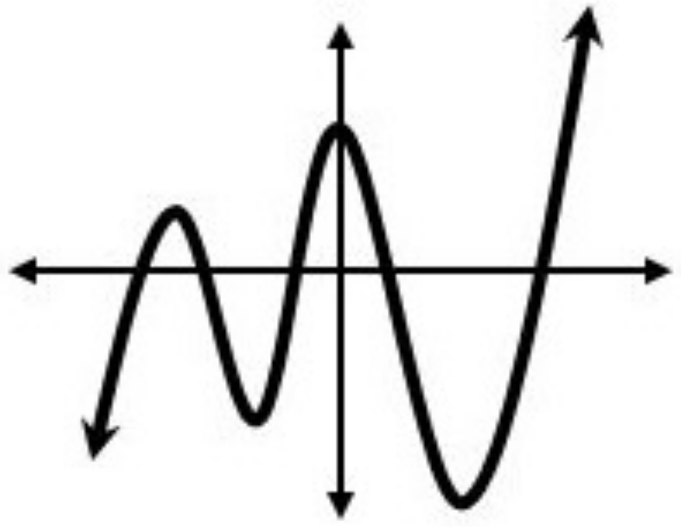
Maths 101: Polynomial functions [2]

Higher degree (with degree of 4): $y = x^4$



The effect of x on y is said to be W-shaped

Higher degree (with degree of 5): $y = x^5$



The effect of x on y is now said to be Wiggly-shaped

You may think to yourself why go through these classes of polynomials?

GLM versus GAM: which one to use? [1]

- GAMs enable the user to fit a polynomial function on an independent variable in order for the model to fit the data nicely.

Hypothetical scenario (simulated data):

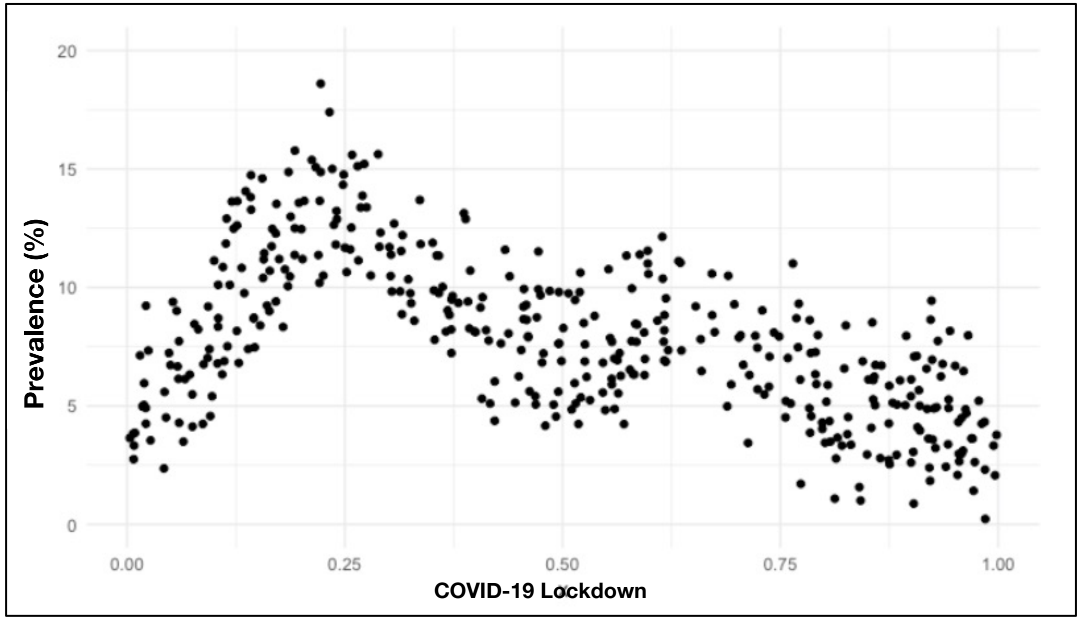
Assessing the impact of COVID-19 lockdown phases and various sociodemographic factors on prevalence of mental health in the British population.

Suppose $x_{i,1}$ represent the time/phase of lockdown, and y_i is the measured prevalence of mental health.

What model should we pick?

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p} + \varepsilon$$

$$y_i = \alpha + f_1(x_{i,1}) + f_2(x_{i,2}) + \cdots + f_p(x_{i,p}) + \varepsilon$$



GLM versus GAM: which one to use? [2]

- GAMs enable the user to fit a polynomial function on an independent variable in order for the model to fit the data nicely.

Hypothetical scenario (simulated data):

Assessing the impact of COVID-19 lockdown and various sociodemographic factors on prevalence of mental health in the British population.

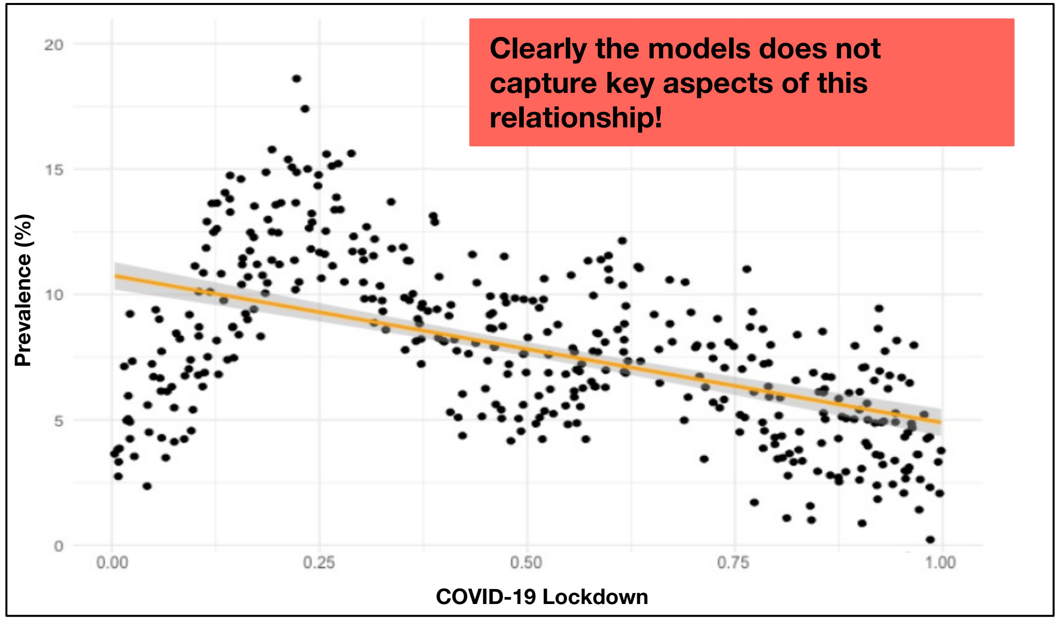
Suppose $x_{i,1}$ represent the time/phase of lockdown, and y_i is the measured prevalence of mental health.

Here, we have regressed $x_{i,1}$ using a linear function

What model should we pick?

$$y_i = \beta_0 + \beta_1 \boxed{x_{i,1}} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p} + \varepsilon$$

$$y_i = \alpha + f_1(x_{i,1}) + f_2(x_{i,2}) + \cdots + f_p(x_{i,p}) + \varepsilon$$



GLM versus GAM: which one to use? [3]

- GAMs enable the user to fit a polynomial function on an independent variable in order for the model to fit the data nicely.

Hypothetical scenario (simulated data):

Assessing the impact of COVID-19 lockdown and various sociodemographic factors on prevalence of mental health in the British population.

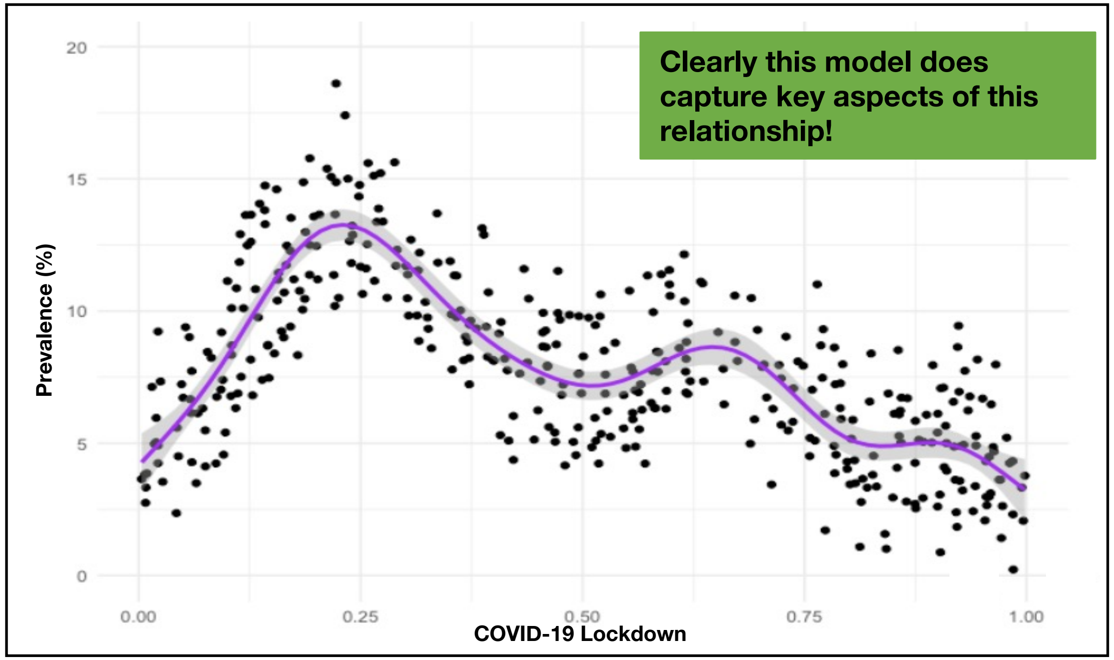
Suppose $x_{i,1}$ represent the time/phase of lockdown, and y_i is the measured prevalence of mental health.

What about we apply some higher degree function & regress it on $x_{i,1}$?

What model should we pick?

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \cdots + \beta_p x_{i,p} + \varepsilon$$

$$y_i = \alpha + \boxed{f_1(x_{i,1})} + f_2(x_{i,2}) + \cdots + f_p(x_{i,p}) + \varepsilon$$



Smooth Spline

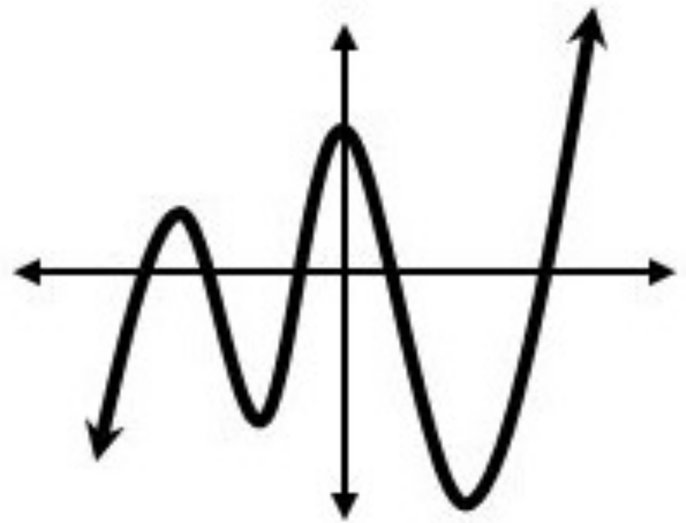
- Note that function $f_1()$ wrapped around our independent variable $x_{i,1}$ is device for smoothing the data.
- Smoother devices can be anything from a quadratic, cubic to something that is of higher degree
- Eyeballing the GAM fit for COVID-19 lockdown variable in relation to prevalence of mental health in Britain – looks something of a function with degree of 5

$$\boldsymbol{y_i = \alpha + \boxed{f_1(x_{i,1})} + f_2(x_{i,2}) + \cdots + f_p(x_{i,p}) + \varepsilon}$$

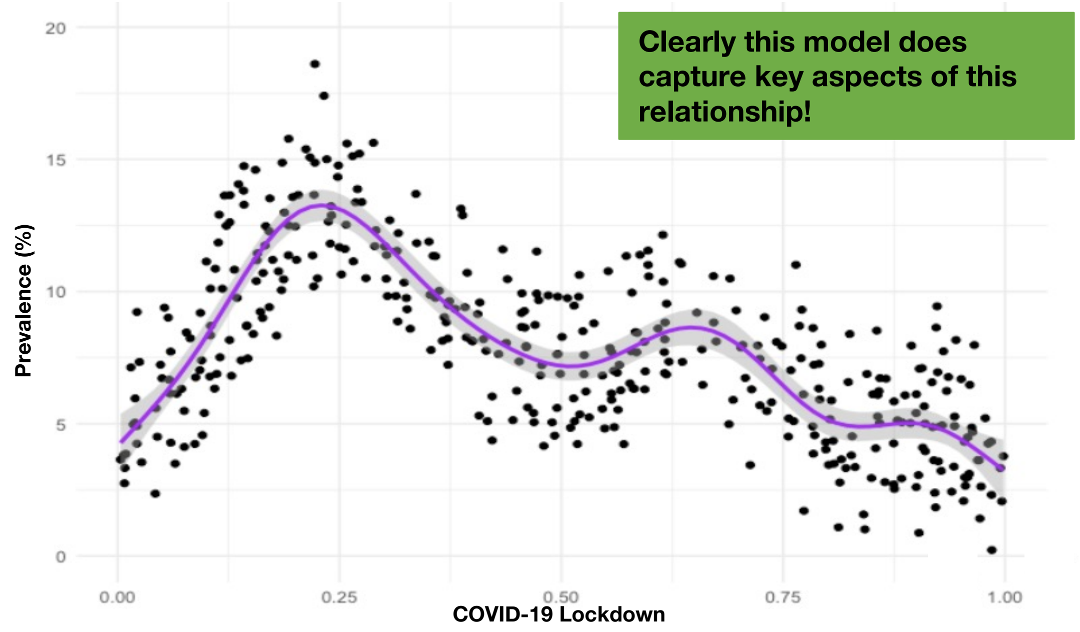
$$f_1(x_{i,1}) = \beta_5 x_{i,1}^5 + \beta_4 x_{i,1}^4 + \beta_3 x_{i,1}^3 + \beta_2 x_{i,1}^2 + \beta_1 x_{i,1}$$

This is a known as a **smooth spline**, that allows for flexibility in the fitting. A series of Basis functions forms a GAM.

Higher degree (with degree of 5): $y = x^5$



The effect of x on y is now said to be Wiggly-shaped



Basis function

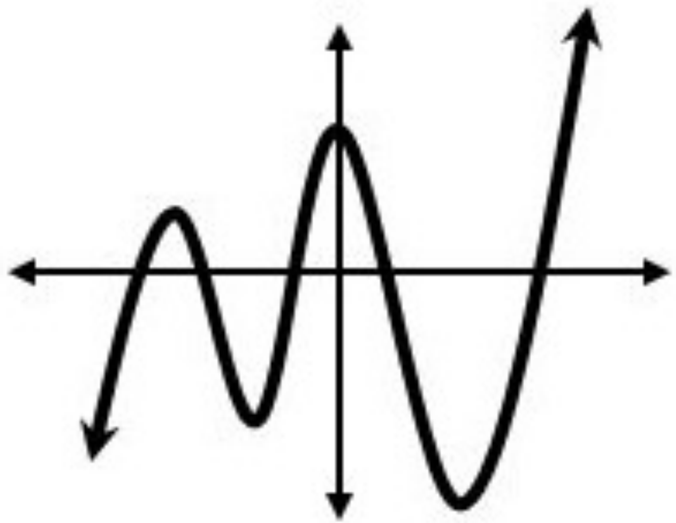
- Note that function $f_1()$ wrapped around our independent variable $x_{i,1}$ is device for smoothing the data.
- Smoother devices can be anything from a quadratic, cubic to something that is of higher degree
- Eyeballing the GAM fit for COVID-19 lockdown variable in relation to prevalence of mental health in Britain – looks something of a function with degree of 5

$$\boldsymbol{y_i = \alpha + f_1(x_{i,1}) + f_2(x_{i,2}) + \cdots + f_p(x_{i,p}) + \varepsilon}$$

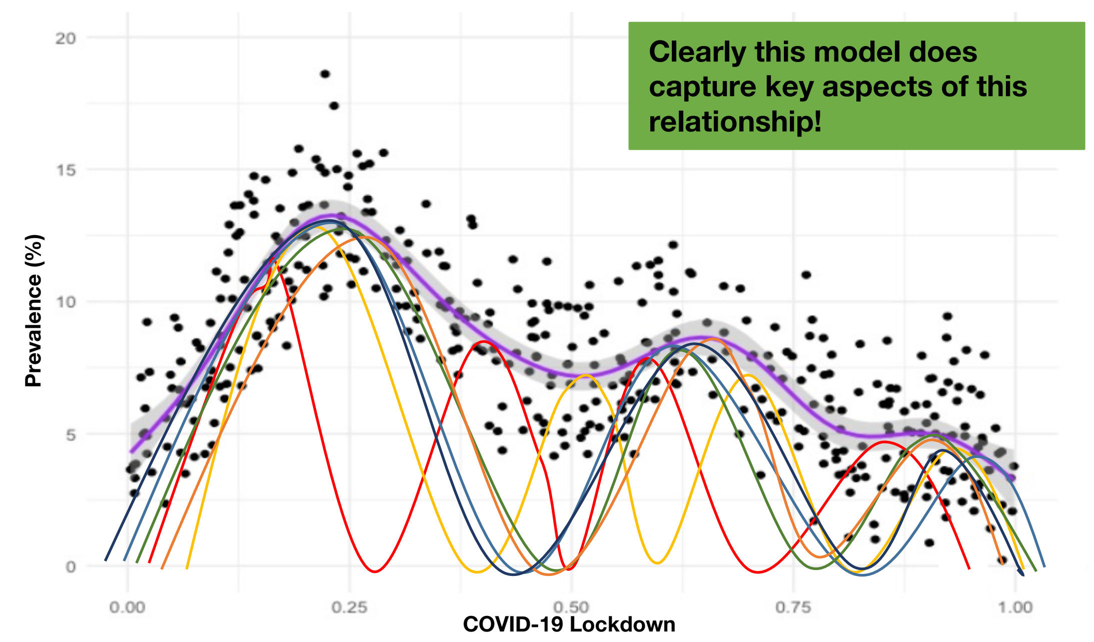
$$f_1(x_{i,1}) = \boxed{\beta_5 x_{i,1}^5 + \beta_4 x_{i,1}^4 + \beta_3 x_{i,1}^3 + \beta_2 x_{i,1}^2 + \beta_1 x_{i,1}}$$

The smoother/splines actually are constructed by many smaller functions, these are called **Basis Functions.** Note - each smooth is a sum of number of Basis functions, and each Basis function is multiplied by a coefficient such that each are a parameter in a model.

Higher degree (with degree of 5): $y = x^5$

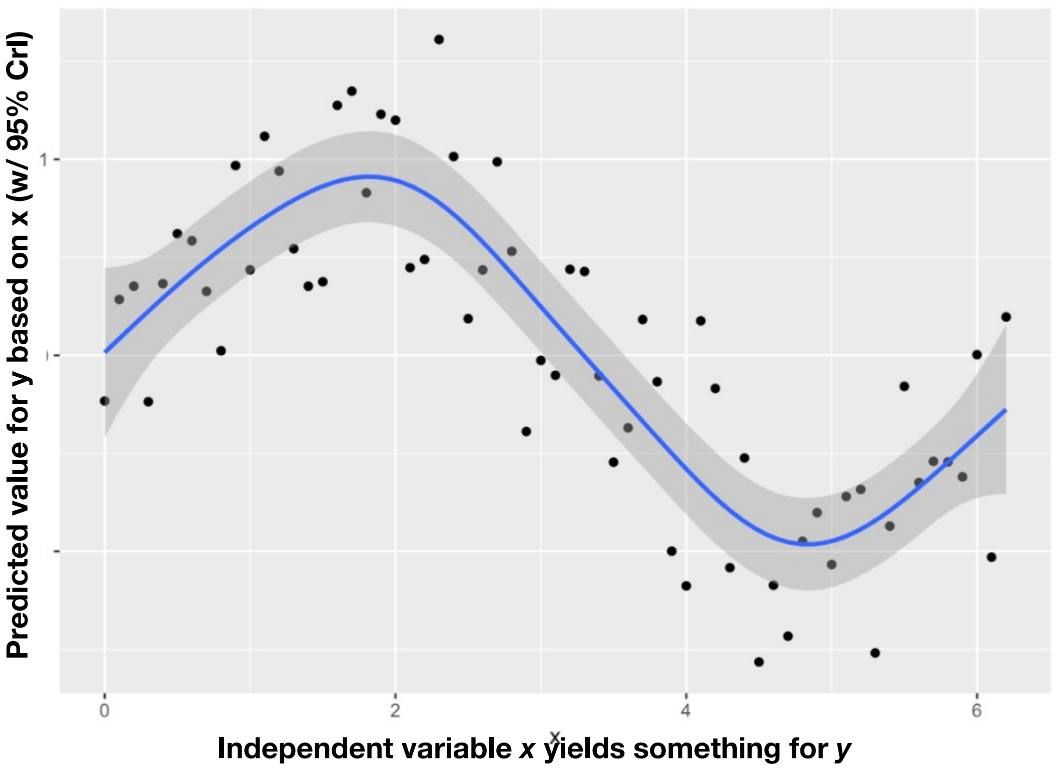


The effect of x on y is now said to be Wiggly-shaped

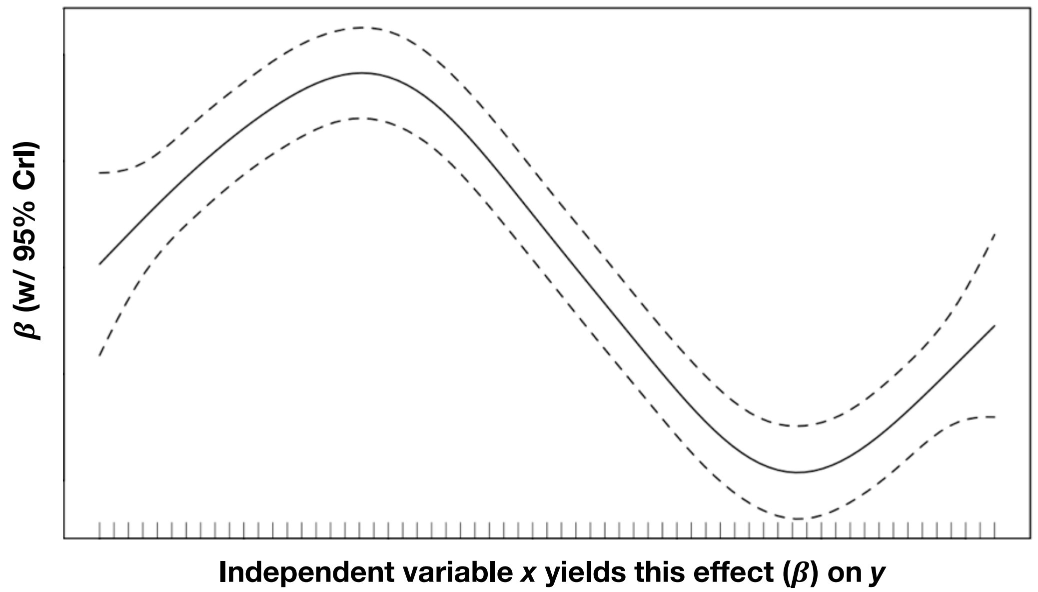


Outputs from the Basis function

A: Model fitting to data for Prediction & Forecast



B: Estimation of Parameter Coefficients



For GAM, the dependent variables can come from a Gaussian, Binomial and/or Poisson distribution. Hence, you can specify the likelihood function accordingly. As for the independent variable – it can also take both continuous and categorical variables.

Example and Interpretation

Example 1: Air quality and Respiratory admissions in Turin Province [1]

GOAL: Assessing the non-linear relationships between these three variables with hospitalisation in Turin.

y_i = Total admission (in an area)
$x_{i,1}$ = particulates in 2.5-10 cubic m (PM10)
$x_{i,2}$ = Nitrogen Dioxide NO_2 (parts per billion)
$x_{i,3}$ = Carbon Monoxide (parts per billion)

Model formulation

- Using a GAM with Basis function all variables

$$y_t = \alpha + f_1(x_{i,1}) + f_2(x_{i,2}) + f_3(x_{i,3}) + \varepsilon$$

- Specify likelihood function. The outcome is counts – thus its Poisson, and we will use the $\log()$ as our link function.

$$y_t \sim \text{Poisson}(\mu_t): \log(\mu_t) = \alpha + f_1(x_{i,1}) + f_2(x_{i,2}) + f_3(x_{i,3})$$

- Build Bayesian model

Recall the Bayes' Rule: $P(\theta|Y) \propto P(Y|\theta)P(\theta)$

$$P(\boldsymbol{\beta}_k, \tau| \log(\mu_t)) \propto P(\log(\mu_t)| \boldsymbol{\beta}_k, \tau)\, P(\boldsymbol{\beta}_k)\, P(\tau)$$

Let's run the analysis in R using Bayesian Regression Modelling in Stan package (brms).
You may think this will be hard – but the next step is super easiest!

Example 1: Air quality and Mortality in Chicago (1987-2000) [2]

R Code from “brms” >>> brms::stancode(model.bayes.gam) >>> Translate to Stan code automatically

```
install.packages(“brms”)
library(“brms”)

# run a GAM model
model.bayes.gam <- brm(bf(Overall ~ s(PM10) + s(CO) + s(NO2)),
                  data = respiratory_data,
                  family = poisson(),
                  prior = prior.list,
                  cores = 6,
                  iter = 8000,
                  warmup = 1000,
                  thin = 10,
                  refresh = 0,
                  control = list(adapt_delta = 0.99))
```

This time around the coding is a lot easier. You can use this as an alternative option to writing raw Stan code.

```
// generated with brms 2.18.0
functions {
}
data {
  int<lower=1> N;  // total number of observations
  int Y[N];  // response variable
  // data for splines
  int Ks;  // number of linear effects
  matrix[N, Ks] Xs;  // design matrix for the linear effects
  // data for spline s(PM10)
  int nb_1;  // number of bases
  int knots_1[nb_1];  // number of knots
  // basis function matrices
  matrix[N, knots_1[1]] Zs_1_1;
  // data for spline s(CO)
  int nb_2;  // number of bases
  int knots_2[nb_2];  // number of knots
  // basis function matrices
  matrix[N, knots_2[1]] Zs_2_1;
  // data for spline s(NO2)
  int nb_3;  // number of bases
  int knots_3[nb_3];  // number of knots
  // basis function matrices
  matrix[N, knots_3[1]] Zs_3_1;
  int prior_only;  // should the likelihood be ignored?
}
transformed data {
}
parameters {
  real Intercept;  // temporary intercept for centered predictors
  vector[Ks] bs;  // spline coefficients
  // parameters for spline s(PM10)
  // standarized spline coefficients
  vector[knots_1[1]] zs_1_1;
  real<lower=0> sds_1_1;  // standard deviations of spline coefficients
  // parameters for spline s(CO)
  // standarized spline coefficients
  vector[knots_2[1]] zs_2_1;
  real<lower=0> sds_2_1;  // standard deviations of spline coefficients
  // parameters for spline s(NO2)
  // standarized spline coefficients
  vector[knots_3[1]] zs_3_1;
  real<lower=0> sds_3_1;  // standard deviations of spline coefficients
}
transformed parameters {
  // actual spline coefficients
  vector[knots_1[1]] s_1_1;
  // actual spline coefficients
  vector[knots_2[1]] s_2_1;
  // actual spline coefficients
  vector[knots_3[1]] s_3_1;
  real lprior = 0;  // prior contributions to the log posterior
  // compute actual spline coefficients
  s_1_1 = sds_1_1 * zs_1_1;
  // compute actual spline coefficients
  s_2_1 = sds_2_1 * zs_2_1;
  // compute actual spline coefficients
  s_3_1 = sds_3_1 * zs_3_1;
  lprior += student_t_lpdf(Intercept | 3, 3, 2.5);
  lprior += student_t_lpdf(sds_1_1 | 3, 0, 2.5)
    - 1 * student_t_lccdf(0 | 3, 0, 2.5);
  lprior += student_t_lpdf(sds_2_1 | 3, 0, 2.5)
    - 1 * student_t_lccdf(0 | 3, 0, 2.5);
  lprior += student_t_lpdf(sds_3_1 | 3, 0, 2.5)
    - 1 * student_t_lccdf(0 | 3, 0, 2.5);
}
model {
  // likelihood including constants
  if (!prior_only) {
    // initialize linear predictor term
    vector[N] mu = rep_vector(0.0, N);
    mu += Intercept + Xs * bs + Zs_1_1 * s_1_1 + Zs_2_1 * s_2_1
+ Zs_3_1 * s_3_1;
    target += poisson_log_lpmf(Y | mu);
  }
  // priors including constants
  target += lprior;
  target += std_normal_lpdf(zs_1_1);
  target += std_normal_lpdf(zs_2_1);
  target += std_normal_lpdf(zs_3_1);
}
generated quantities {
  // actual population-level intercept
  real b_Intercept = Intercept;
}
```

Example 1: Air quality and Mortality in Chicago (1987-2000) [2]

Smoothed terms

Variables	Smoothed term (95% Credibility)	Convergence ($\hat{R}$)
PM10	6.57 (3.92 to 11.63)	1.01 < 1.05
NO2	6.30 (4.00 to 10.03)	1.01 < 1.05
CO	5.83 (3.72 to 9.77)	1.01 < 1.05

Meaning: this is the variance parameter, which has the effect of controlling the "wiggliness" of the smooth — the larger this value the more wiggly the smooth. We can see that the credible interval doesn't include 0 so there is evidence that a smooth is required over and above a linear.

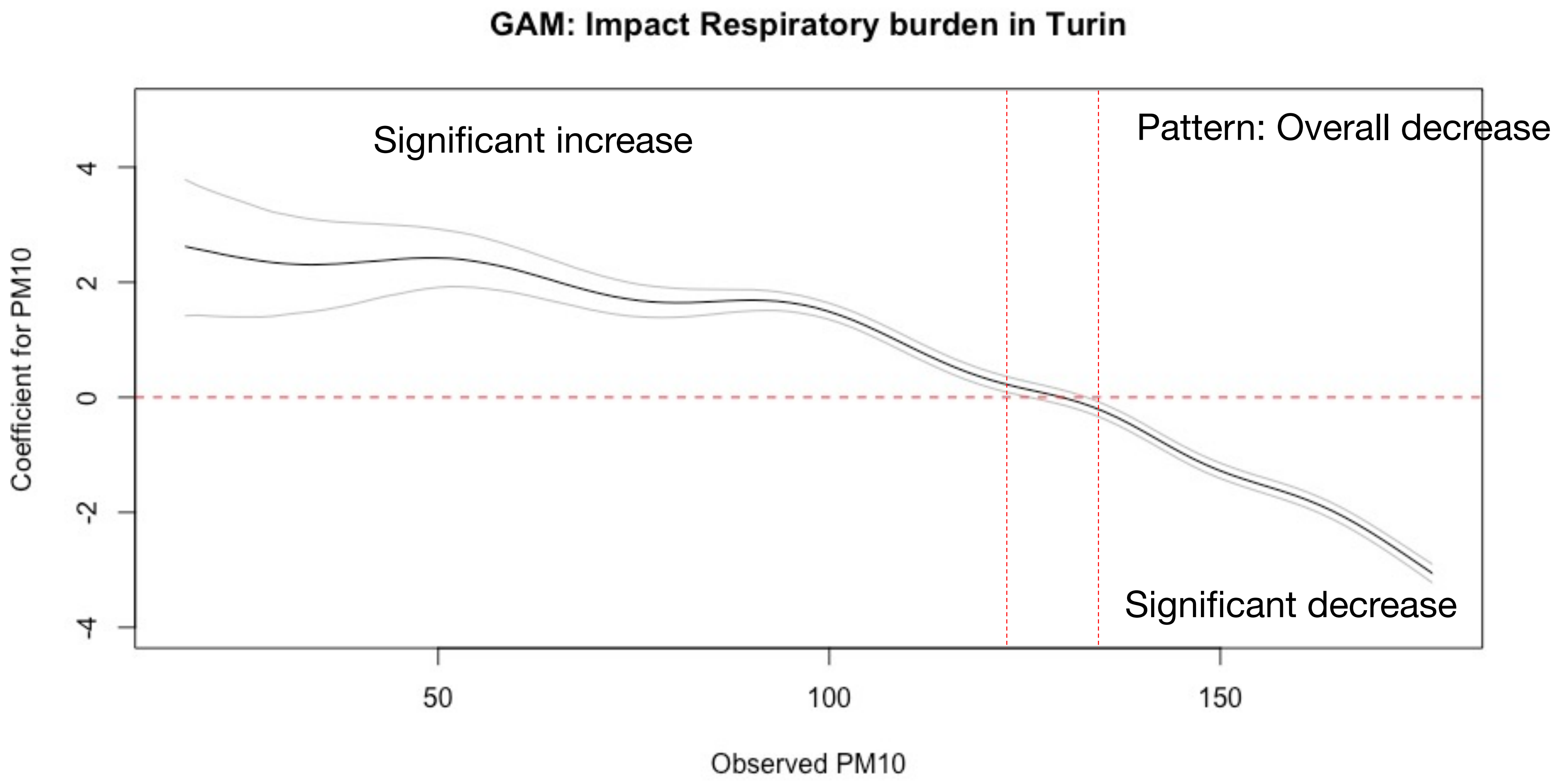
Here, it was correct for us to apply a GAM model on these three variables. Also, the model is valid since the $\hat{R}$ estimates are below 1.05

Population-level (Global) effect

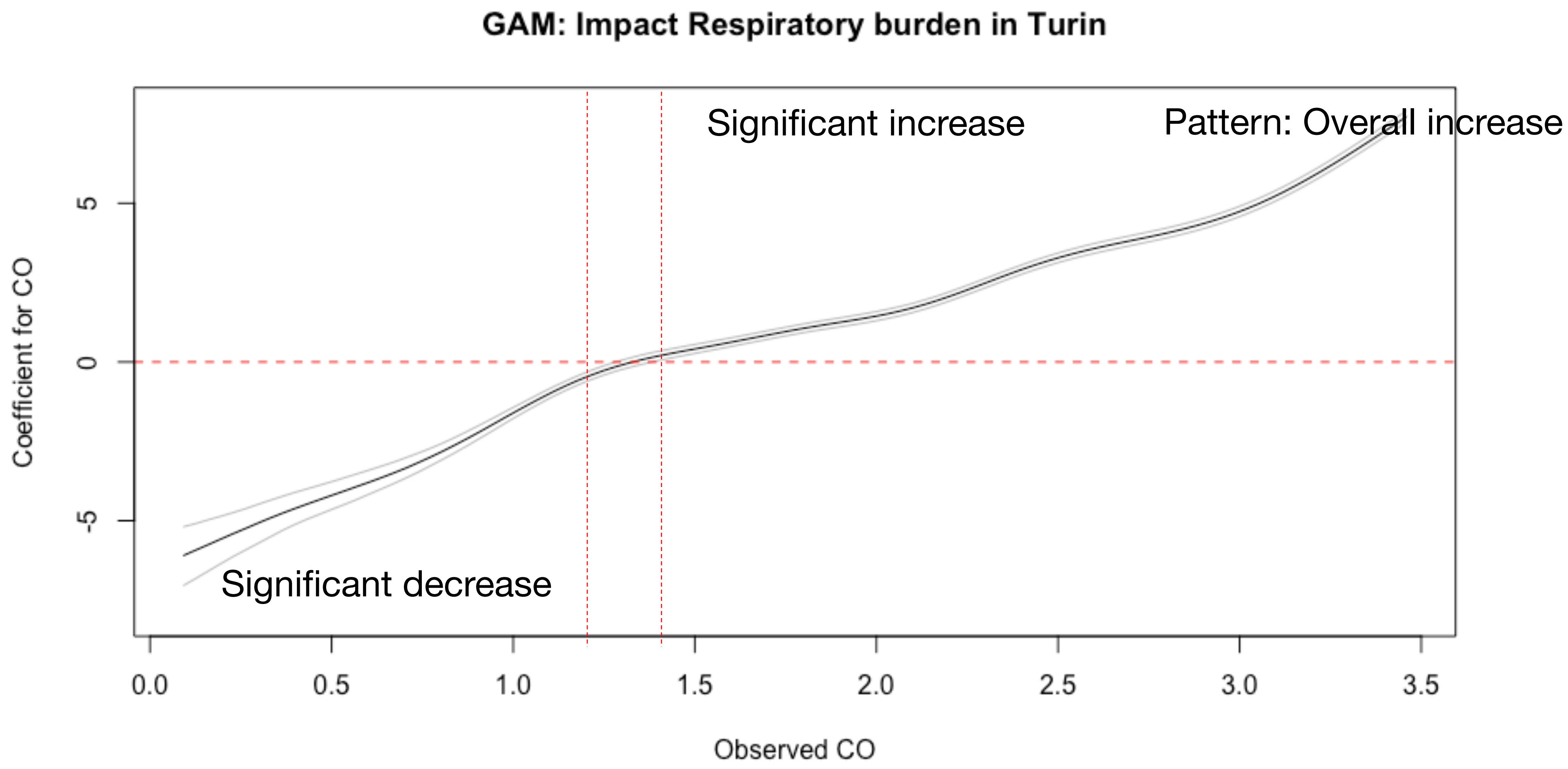
Variables	Coefficient (95% Credibility)	Convergence ($\hat{R}$)
Intercept	3.56 (3.54 to 3.59)	1.00 < 1.05
PM10	-19.46 (-26.25 to -12.94)	1.01 < 1.05
NO2	4.35 (-5.43 to 14.39)	1.01 < 1.05
CO	33.75 (27.43 to 39.90)	1.01 < 1.05

Meaning: These are our global estimates which are considered as fixed effects. We will interpret these as we usually interpret a regression the usual way.

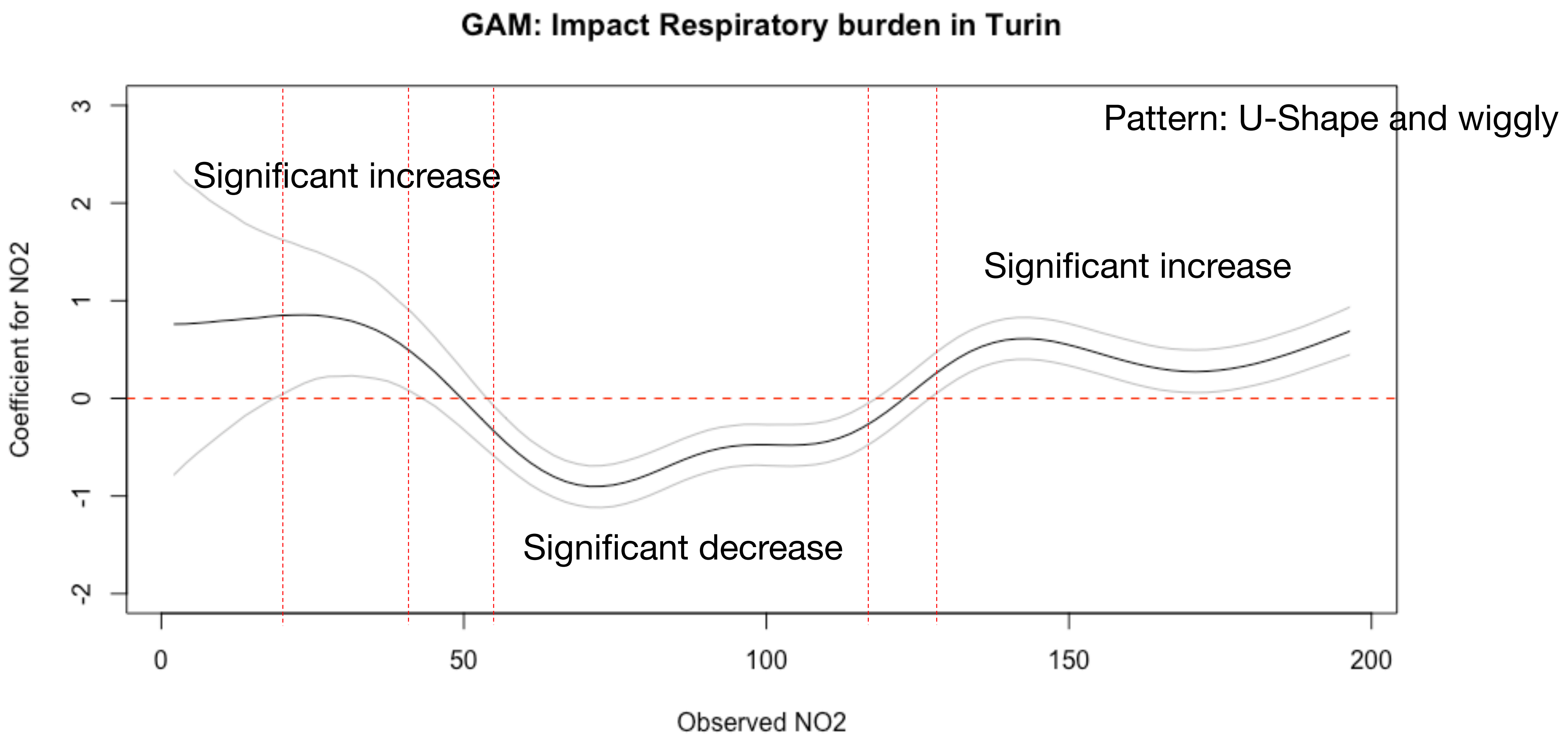
Population-level (Global) effect of PM10: -19.46 (-26.25 to -12.94)



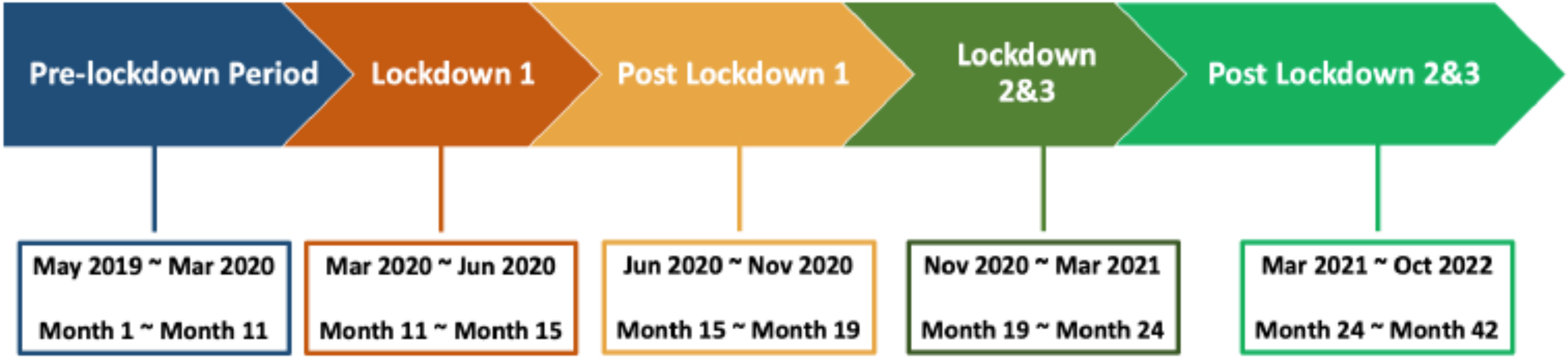
Population-level (Global) effect of CO: 33.75 (27.43 to 39.90)



Population-level (Global) effect of NO2: 4.35 (-5.43 to 14.39)



Example 2: COVID-19 lockdown phases and impact on crime resurgence in Nottingham (Antisocial Behaviour [ASB]) [1]



- Understand the geospatial crime recovery patterns for the incidence of Antisocial Behaviour (ASB) events in-between the different lockdown phases in Nottingham across all 182 LSOAs.
- To examine whether these ASB patterns were statistically significant. Hence a stratified Poisson GAM model was used for each LSOA.

Gehui Qi (2023) Investigating the Crime Recovery Patterns in Nottingham in the Post-Lockdown Period using Social Disorganisation Theory. UCL Undergraduate Dissertation 2022/23. Department of Geography. Submitted BA Geography with Social Data Sciences. **Download:** [Click]

Data Repository: https://github.com/qghuihuihui/Dissertation-2023

Example 2: COVID-19 lockdown phases and impact on crime resurgence in Nottingham (Antisocial Behaviour [ASB]) [2]

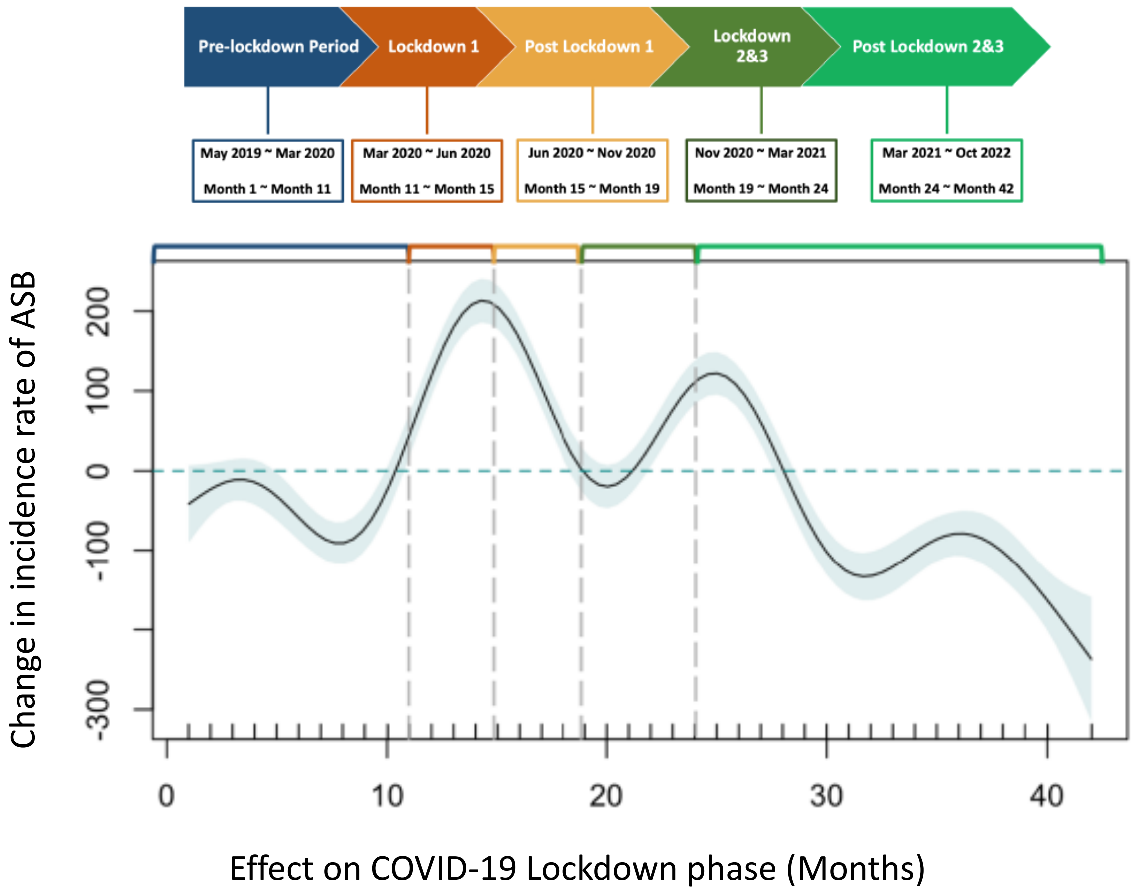
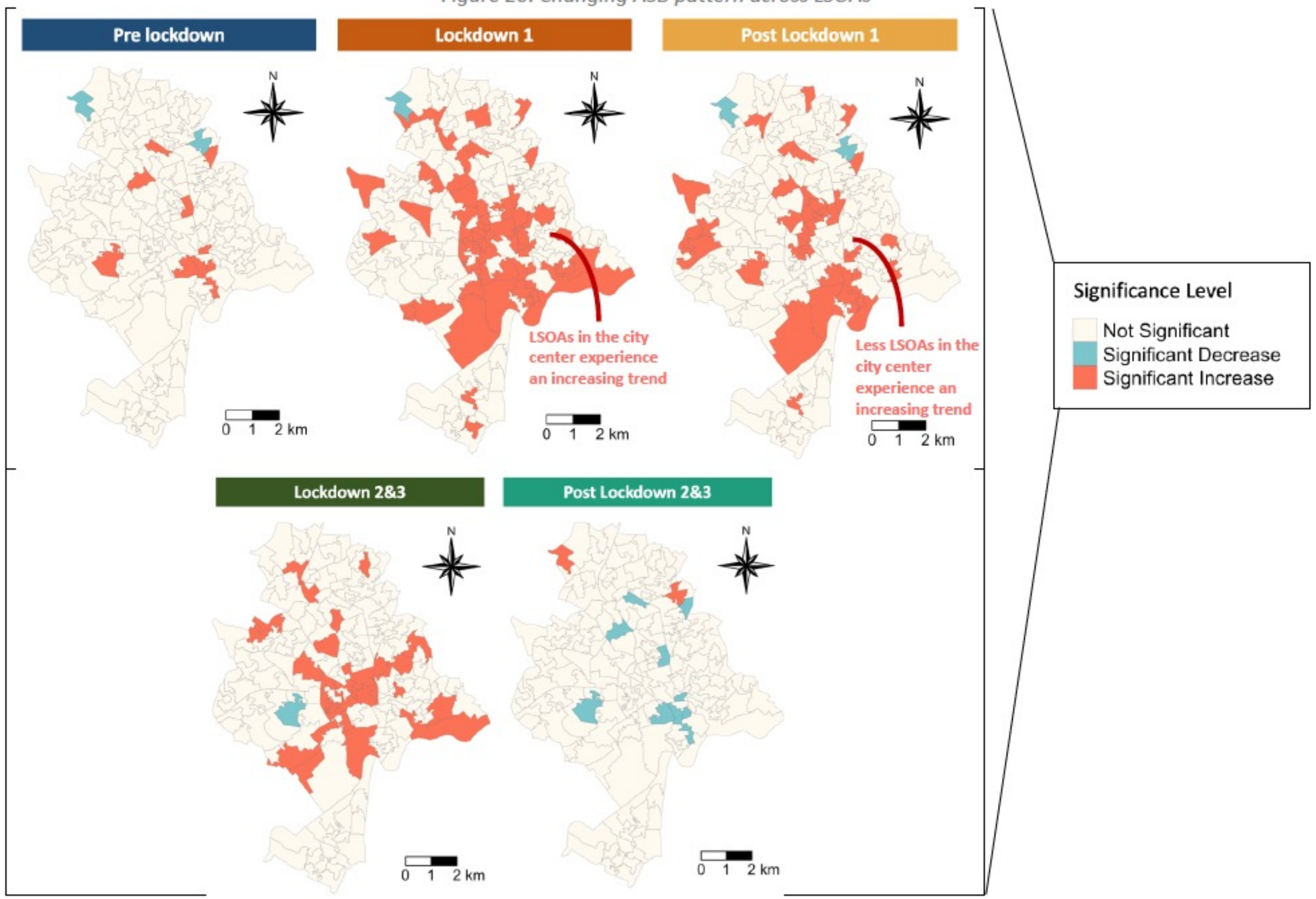


Figure 20: Changing ASB pattern across LSOAs



A series of Poisson GAMs were implemented on each LSOAs. 182 GAM plots were generated for each LSOA which had patterns showing the effect of lockdown (months) on changes in incidences of ASB.

The phases for LSOAs were broadly classed either a significant 'increase', 'decrease' or not significant all in ASB during those periods. These were extracted and plotted on to the map of Nottingham for interpretation.

Any questions?