Advanced Topics in Regression

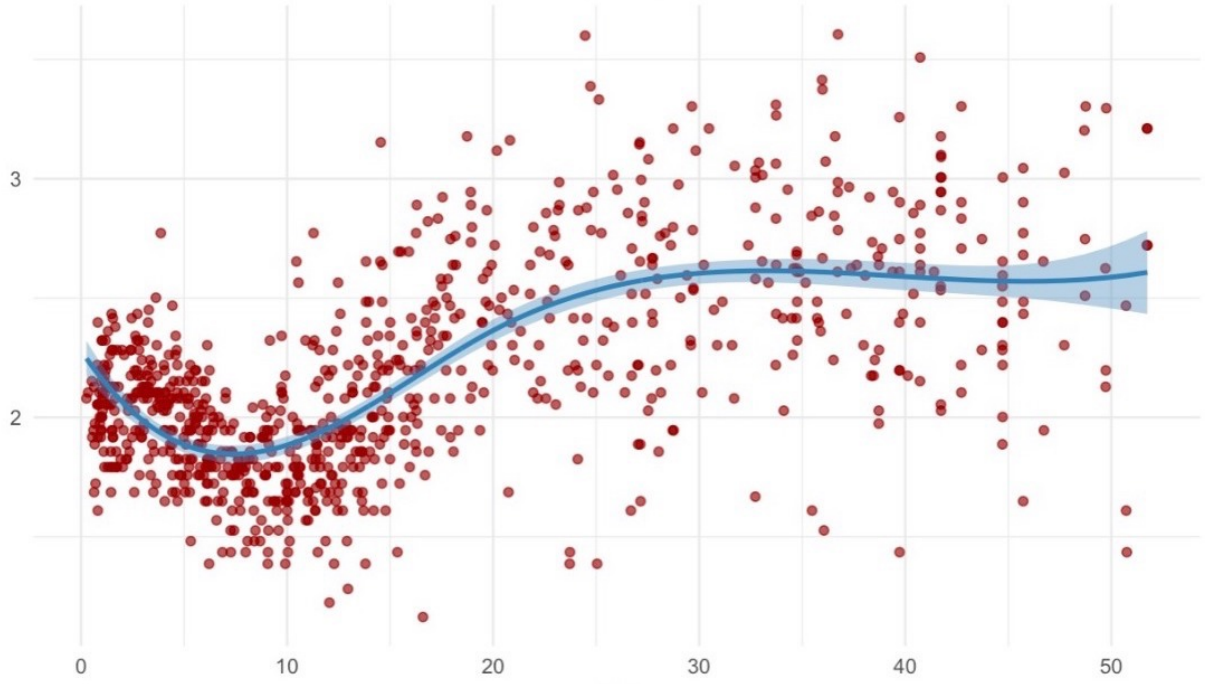
Summary: Section A

Module Purpose

This section covers single & multi-dimensional Splines, GAMs, GLMs, MARS, Wavelets & Functional Data Analysis.

These techniques are designed to model non-linear data & are somewhat semi-parametric (not as limited to known functional forms as in parametric models but still offering some interpretability - unlike non-parametric models). Constantly tradeoff fit vs smooth.

This document focuses on answering the *why* question - the notes deal with *what*.



Univariate Splines

Transformations can't seem to **linearise the data** or **remove heteroscedasticity**.

Long Term trends (time series & otherwise) are seldom stationary.

Smoothness vs Variability Explained

Regression Models

Simple Regression: simplest model but most restricted

Polynomial Regression: still requires known/specified functional form (we require more automatic estimation)

Broken Stick Model: provides a change in slope at a known point

Piecewise Polynomial: can smooth well (continuous at second derivative)

Regression Splines

Weighted sum of basis functions is the smooth curve.

Cubic Splines

A spline is a sequence of piecewise functions (often polynomials) (each of order M), joined at knots, continuous up to $(M-2)$th derivative at each knot. Splines are still linear models (linear in parameters). Knots primarily determine the wiggliness of the curve.

Extrapolating past the range of the data is dangerous for both polynomial models & splines, because predictive forecasts are made on few datapoint on the extremes. Natural cubic splines (linear beyond boundary knots) reduce some of this variation in the tails.

B Splines

A space of spline functions of a particular order & knot sequence is a vector space. There are many equivalent bases for representing the vector space (different set of vectors can be used to define the same vector space). B Splines has local basis functions - reducing collinearity & improving computational stability. Sparse basis matrix. Given a basis design matrix:

$$\hat{\beta} = (X'X)^{-1}X'Y$$

$$\hat{f}(x) = X\hat{\beta} = X(X'X)^{-1}X'Y = HY$$

$$Var(\hat{\beta}) = \hat{\sigma}^2(X'X)^{-1} = \Sigma$$

$$Var(\hat{f}(x)) = Var(X\hat{\beta}) = X\Sigma X'$$

Basis Functions for Periodic Data

Cyclical data, end and start should equal. Traditionally modeled through Fourier Series (sinusoidal functions). Regression with trig terms = harmonic regression. To fit a cyclical/periodic spline start with cubic splines basis & add constants to make spline continuous up to 2nd derivative at the cycle end/start.

Radial Basis Functions

Used for thin plate splines - radially symmetric basis. Useful for multivariate data: convert a multivariate problem to a univariate one.

Summary

The aim is to find a flexible f(x) global polynomial (using splines or piecewise polynomials). Calculate basis & fit regular regression procedure (linear or generalised linear). Learn a flexible, non-linear relationship between X and y.

Penalized Splines

It's difficult to know how many/where knots should go. Model selection involved dropping knots/ many models - AIC, Mallows CP not suitable (why? Rely on functional form?). Knot selection requires experience & time. A more flexible & automatic approach is required to appropriately trade off fit & smooth.

Penalized (Regression) Splines

Takes away the problem of selecting number & placement of knots. Choose a large enough number of knots but constrain their influence. Often end up with less basis than data points. (Same idea with Wavelets).

Control Model Complexity

- Restrict number of terms in the model (knots)
- *Selection methods*: only select variables or basis functions that contribute to the model (subset selection). Including MARS.
- *Regularization methods*: use entire dictionary but restrict coefficients (eg Ridge Reg). Adding information to solve an ill-posed problem. One can consider Bayesian statistics regularization adding info through the prior.

Splines context

Simpler models are smoother. We can smooth by penalising large Betas

$$\sum \beta_j^2 \leq C$$

i.e. we need to choose $\boldsymbol{\beta}$ to minimize the error or residual sum of squares

$$||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 \qquad \text{subject to} \qquad \boldsymbol{\beta}'\mathbf{D}\boldsymbol{\beta} \leq C.$$

We can incorporate constraints into minimization or maximization problems by using a Langrange multiplier λ and then minimizing the *penalized error sum of squares*:

$$||\mathbf{y} - \mathbf{X}\boldsymbol{\beta}||^2 + \lambda\boldsymbol{\beta}'\mathbf{D}\boldsymbol{\beta}.$$

In general, minimize: $PRSS = \sum (y_i - f(x_i))^2 + \lambda J(f)$

$f(x_i)$ how well data is explained

$J(f)$ regularisation / penalty term

λ smoothing parameter.

$\lambda = 0$ any function that interpolates the data

$\lambda = \infty$ simple least squares line (no 2nd derivative can be tolerated)

Intercept term (& any global basis function) is not penalized.

Solution: $\hat{y} = x(X'X + \lambda D)^{-1}X'y$

Penalized (Regression) Splines: 2nd Approach

$f'(x)$ slope

$f''(x)$ change in slope

$\int f''(x)^2 dx$ area under change in square (absolute but amplified) slope > wiggliness

When minimising PSS λ is fixed.

P Splines: Adjacent curves more similar == smoother.

Smoothing Splines

Find any function to minimise:

$$PRSS = \sum (y_i - f(x_i))^2 + \lambda \int f''(x)^2 dx$$

Fundamentally an Optimization problem. The solution is natural cubic splines or thin plate splines (dimensionality).

$$f(x) = \sum_{i=1}^{n} N_j(x)\beta_j, \qquad N_j(x) : \textit{natural} \text{ spline basis}$$

$$\text{minimize: } (\mathbf{y} - N\beta)^T(\mathbf{y} - N\beta) + \lambda\beta^T\Omega\beta$$

$$\hat{\beta} = (N^T N + \lambda\Omega)^{-1} N^T \mathbf{y}$$

$$\hat{f}(x) = N(N^T N + \lambda\Omega)^{-1} N^T \mathbf{y} = \mathbf{S}_\lambda \mathbf{y}$$

S_λ is a smoothing matrix dependent on λ, analogous to the HAT matrix. Still linear in that $\hat{y} = S_\lambda y$. $Tr(S_\lambda)$ gives the effective degrees of freedom (number of parameter estimates) used.

In practice, especially if n is large, it is more computationally efficient to use penalized regression splines.

Choosing Smoothing Parameter λ

Measure Performance of a Smoother

MSE

expected squared distance between an estimator and the true underlying parameter

$$MSE(\hat{\theta}) = E\left[(\hat{\theta} - \theta)^2\right]$$

At a specific x value:

$$\begin{aligned} MSE[\hat{f}(x)] &= E\left[\{\hat{f}(x) - f(x)\}^2\right] \\ &= \left[E\left(\hat{f}(x)\right) - f(x)\right]^2 + Var(\hat{f}(x)) \\ &= \text{Bias}^2 + \text{ Variance} \end{aligned}$$

MSPE

Measures the quality of a predictor of an observation: expected squared distance between prediction & observation (includes individual variation whilst MSE only includes average variance).

$$\begin{aligned} MSPE[\hat{f}(x^*)] &= E[(Y - \hat{f}(x^*))^2] \\ &= Var(Y) + MSE(\hat{f}(x^*)) \end{aligned}$$

Distinction between MSE & MSPE is analogous to Confidence Intervals vs Predictive Intervals. MSPE involves observations AND their deviation from expected value. These metrics measure a single point.

EPE - Expected Prediction Error

MSPE averaged over all observations. We don't know f(x) the true function so we can test on a train/test split.

$$EPE(\hat{f}) = E\left[(Y - \hat{f}(X))^2\right]$$

LOOCV

Gives an estimate of EPE (mean MSPE) without testing data.

Generalised Cross Validation (GCV)

An approximation for LOOCV without repeated computation. GCV alleviates the tendency of cross validation to under-smooth. Trace(S) is the effective number of parameters. GCV is used to select λ such that average predicted error is minimised.

$$GCV(\hat{f}) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{f}(x_i)}{1 - trace(S)/N}\right)^2$$

Multivariate Splines

Regression

- Models linear (in parameters) relationships
- Everything is interpretable
- Models are mostly additive
- Can add 2 way interaction terms (can add more though it introduces other problems, multicollinearity etc).

Neural Networks

- Non-interpretable parameters
- Great at modelling non-linear relationships (non-linear relationships & nonlinear combinations)
- Great at modelling interaction effects

Multidimensional Splines

- More interpretability than ANNs, more flexible than standard regression models
- Able to add interaction surface to regression model
- Isotropic (uniformity in all dimensions) use Thin Plate Splines, if covariates have different scales use Tensor Product Smooths.
- Thin Plate Splines: euclidean distance based radial basis
- Tensor Product Splines: tensor product of separate basis'
- Primarily used for modelling interaction & smoothing spatial data or images

Tensor Product Splines

- For computational ease, multiplication - not tensor product - of basis pairs is used (diagonal)
- Supports variate covariate scales

Thin Plate Splines

- Solution to optimization problem in 2-d
- Computationally expensive to use all datapoint as knots, instead choose a large enough number of knots & then use GCV to determine λ
- 'mgcv' package, Wood selects a large number of equally spaced knots, cuts them back to the sample space, and then uses an eigenvalue decomposition to select a low-rank thin-plate spline basis
- TPS does not require specifying functional form
- Node (knot) number & placement is problematic in cubic splines is essentially automated in thin plate splines.
- Thin plate splines are low rank isotropic smoothers of any number of covariates. By isotropic is meant that rotation of the covariate co- ordinate system will not change the result of smoothing. By low rank is meant that they have far fewer coefficients than there are data to smooth.

GAMs: Generalized Additive Models

GAMs provide a flexible, automatic way to model non-linear relation- ships. Compared to linear models they are often less biased. Even though the coefficients cannot be interpreted, the individual additive terms are easy to interpret and it is easy to understand contributions of individual variables.

Linear Models

$$Y_i \sim N(\mu_i, \sigma^2), \qquad \mu_i = \sum \beta_j x_j$$

- Have the above form
- 'Linear' in parameters (BX) / 'linear' in covariates.

Generalized Linear Models

$$Y_i \sim [\mu_i, \theta], \qquad g(\mu_i) = \sum \beta_j x_j$$

- g(.) link function
- Relax the linear relationship between x-variables & the response
- Y ~ [μ,ø] can denote any probability distribution
- BX is the linear predictor
- Common link functions:

Model	Random	Link	Systematic
Linear Regression	Normal	Identity	Continuous
ANOVA	Normal	Identity	Categorical
ANCOVA	Normal	Identity	Mixed
Logistic Regression	Binomial	Logit	Mixed
Loglinear	Poisson	Log	Categorical
Poisson Regression	Poisson	Log	Mixed
Multinomial response	Multinomial	Generalized Logit	Mixed

Generalized Additive Models

- Relax linearity
- Instead of requiring that the linear predictor is linear in explanatory variables, we allow any form for this relationship with the x-variables, and only require that the relationship of $g(\mu_i)$ to the x-variables is additive

What this means is that we replace (generalized) linear models of the form

$$g(\mu_i) = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_p x_{ip}$$

with (smooth) non-linear functions $f_j(x_{ij})$:

$$g(\mu_i) = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + f_3(x_{i3}, x_{i4}) + \ldots + f_p(x_{ip})$$

- f(.) are usually splines but could be anything
- GLMs are a special case of GAMs
- Also consider partial response functions

GLM > GAMS

We can no longer just view coefficients & p-values as coefficients are no longer interpretable & multiple coefficients belong to one smooth. How do we handle penalisation of multiple terms? How do we fit GAMS?

Additivity

Additivity means smooth terms are added with no interaction terms. We can incorporate interaction terms (as done with 2-dim splines) however the key purpose of a GAM model to understand the effect of predictor variables on the response.

An **additive model** is in-between the linear model and the fully non-parametric approach: we are dropping linearity, dropping strictly parametric functions, but much of the theory, interpretability of linear models can be retained

Fit Models

- OLS
- Penalized Least Squares
- Backfitting

MARS: Multivariate Additive Regression Splines

MARS is an algorithm that builds a piecewise linear additive model. MARS can model continuous response variables or connect to the response via a link function (as in the case of GLMs). MARS are continuous at knots (a prerequisite for splines) as such are considered smooth. Interaction terms (usually limited to 2 covariates) are easily added.

Useful Attributes

- Interpretability
- Large number of predictors
- Interpretable interaction effects.
- Good at selecting which covariates are important
- Non-parametric in that Y~X function form is not requires
- Computationally easy
- Smooth (compared to piecewise regression)

MARS vs Other Models

- Works well with large p (many features).
- Better than regression at capturing interactions & non-linearities.
- Can handle both classification & regression problems.
- Regression tree models lack continuity - MARS are more interpretable
- Basis functions are only non-zero over a limited range, the algorithm picks out small regions in multivariate space where an adjustment is needed to obtain a better model (MARS does not suffer from the curse of dimensionality as severely as other models). MARS thus does not have to use many parameters for every term (compared to a more traditional spline term).
- Parameters are spent only where needed.

There are two tuning parameters:

- degree of interaction allowed (usually = 2)
- λ size of the model (parameters after pruning)

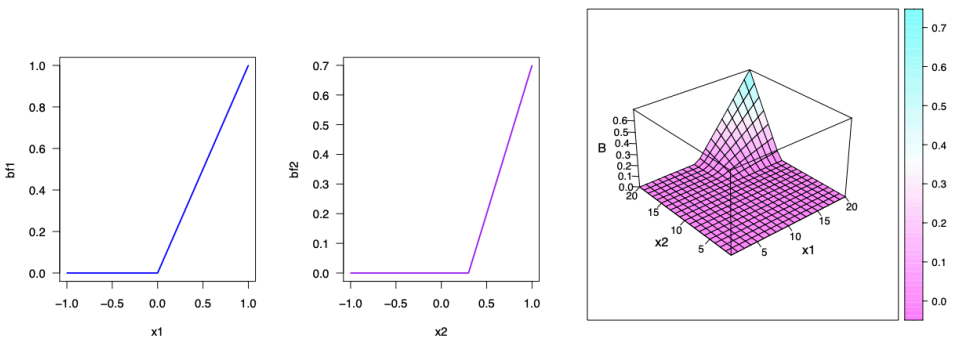
λ the number of basis functions (terms)

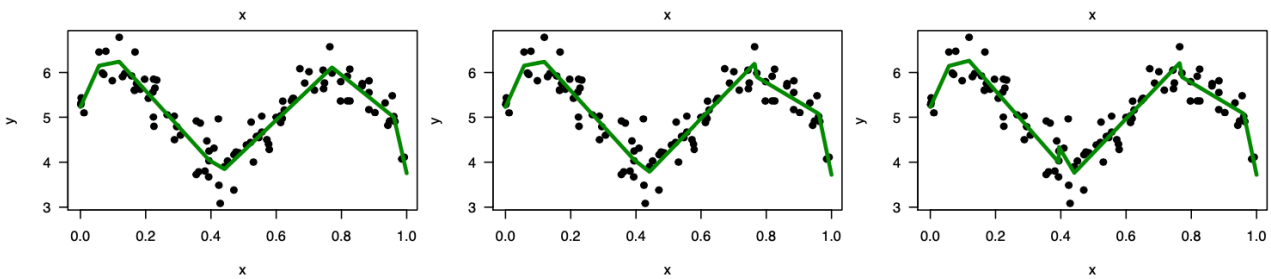
$M(\lambda)$ effective number of parameters

The optimal λ can be chosen as the one with the smallest generalized cross-validation value:

$$GCV(\lambda) = \frac{1}{N}\frac{\sum(y_i - \hat{f}(x_i))^2}{(1 - M(\lambda)/N)^2}$$

GCV can be thought of average squared residual of the model times a penalty to account for the increase in *variance* associated with increase in *model complexity* (number of basis functions λ).





Wavelet Smoothing

A set of wavelets are computed by Dilations (scaling) & Translations (shifting) a wave function & convolving these daughter wavelets with a signal. Different from Fourier transforms in that a FT is localised in frequency but not time (assumes stationarity) whilst wavelets are localised in scale (frequency) and time. Thus wavelets extract information about frequencies & when they occur - better able to detect 'spikes'.

Wavelets are much better at picking up drift, trend, abrupt changes, beginning and end of events, discontinuities or sharp edges.

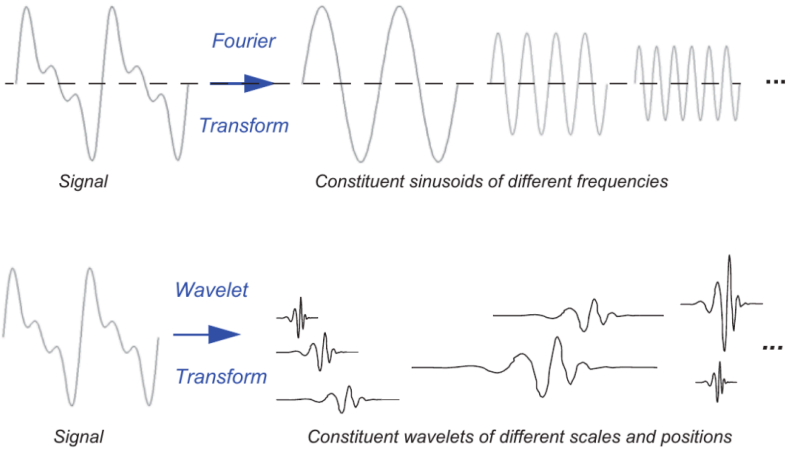
Wavelet transformations are reversible.

Thresholding

One of the main goals of a wavelet analysis is to compress a signal, setting many coefficients to zero achieves this. For smoothing splines we start with a complete basis and then the coefficients are shrunk. For wavelet analysis we also use a complete basis. In addition the basis is orthonormal. We then shrink and select the coefficients and end up with a sparse representation of the original signal. Setting small coefficients equal to zero is removing small detail, or detail which looks like noise.

Applications

- Image (data) compression
- Cleaning noisy data/ remove noise
- Cleaning sound recordings
- Smoothing
- Good for detecting sudden changes in function/image



FDA: Functional Data Analysis

In FDA observation (X or Y or both) are functions - or at least a large number of observations from an underlying - often continuous - function. Functional data are often high-resolution (frequently noisy) data/observations on a continuous underlying process. In one of the steps of FDA, we summarise/rewrite each set of noisy observations as a continuous function.

FDA is used for

- EDA (mean functions, variance functions, cross-correlation)
- Feature alignment & Landmark registration
- Functional principal components
- Functional (linear) model
- Sometimes derivatives are of interest (change in growth, growth spurts, acceleration).

Functional Linear Models

X or Y or both be functions. Parameters/regression coefficients are always functions.

FANOVA Y (function) ~ X (categorical)

Functional Regression

- Function-on-scalar Y (function) ~ X (scalar)
- Scalar-on-function Y (scalar) ~ X (function)

p >> n:

functional regression deals with this by penalising β coefficients into a smooth function (i.e splines), thus we only need to estimate q < n << p parameters

Parameter Estimation

Minimize

$$SSE = \sum_i \int (Y_i(t) - Z_i\beta(t))^2 \, dt = \sum_i \int e_i(t)^2 dt$$

- Pointwise minimization (at each t)
- Basis expansion: model coefficients of basis expansion