Local regression and `loess`

Local regression

- If $f(x)$ is a derivable function in x_0 then, the Taylor's approximation says that it is locally approximated by a line passing through $(x_0, f(x_0))$, i.e.,

$$f(x) = \underbrace{f(x_0)}_{\alpha} + \underbrace{f'(x_0)}_{\beta}(x - x_0) + \text{error}$$

- we introduce the weighted least squares by weighting observations x_i with their distance from x_0:

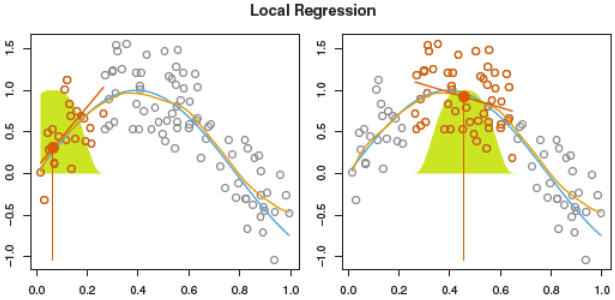
$$\min_{\alpha,\beta} \sum_{i=1}^{n} \left\{ y_i - \alpha - \beta(x_i - x_0) \right\}^2 w_h(x_i - x_0)$$

  - h ($h > 0$) is a scale factor, called bandwidth or smoothing parameter, and
  - $w_h(\cdot)$ is a symmetric density function around 0, said kernel.

Local regression

- By varying x_0, we obtain a whole estimated curve $\hat{f}(x)$.
- the most important component is h, which regulates the smoothness of the curve, while the choice of w is less relevant.
- we could think to w as the density of the normal distribution $N(0, 1)$

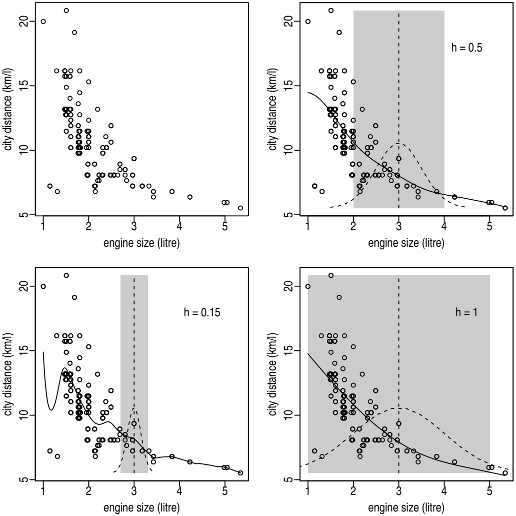
Local regression



Local regression: blue curve represents the real $f(x)$, orange curve corresponds to the local regression estimate $\hat{f}(x)$. The orange points are local to the target point x_0, represented by the orange vertical line. The green bell-shape superimposed on the plot indicates weights assigned to each point, decreasing to zero with distance from the target point. The fit $\hat{f}(x)$ at x_0 is obtained by fitting a weighted linear regression (orange line segment), and using the fitted value at x_0 (orange solid dot) as the estimate $\hat{f}(x_0)$

Local regression

The effect of h is relevant



Variable bandwidths and `loess`

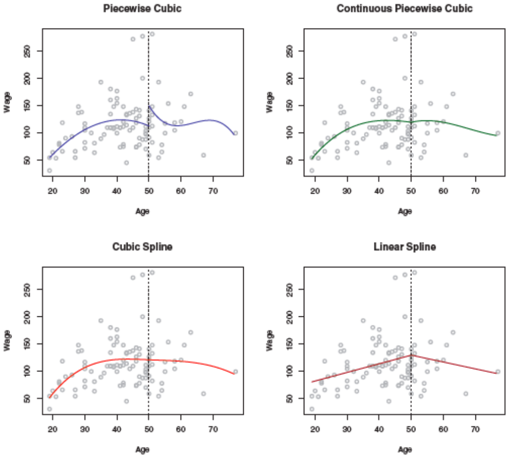
- In many cases, there is an advantage in using a non constant bandwidth along the x-axis, according it to the level of sparseness of observed points
- variable bandwidth: it is reasonable to use larger values of h when x_i are more scattered
- How do we modify h?
- loess: express the smoothing parameter defining the fraction of effective observations for estimating $f(x)$ at a certain point x_0 on the x-axis;
- this fraction is kept constant
- this implies automatically a setting of the bandwidth related to the sparsity of data

Splines

Interpolating splines

- 'Spline' is a mathematical tool useful in many contexts finalised to approximate functions or to interpolate data.
- we choose K points $\xi_1 < \xi_2 < \cdots < \xi_K$, called knots, along the x-axis.
- a function $f(x)$ is constructed, so that it passes exactly through the knots and is free at the other points
- we look for "smooth" functions
- between two successive knots, in the interval (ξ_i, ξ_{i+1}), curve $f(x)$ coincides with a suitable polynomial, of prefixed degree d
- these sections of polynomials meet at point ξ_i $(i = 2, \ldots, K - 1)$
- in the sense that the resulting function $f(x)$ has a continuous derivative from degree 0 to degree $d - 1$ in each of the ξ_i.

Interpolating splines



Top Left: The cubic polynomials are unconstrained. Top Right: The cubic polynomials are constrained to be continuous at age=50. Bottom Left: The cubic polynomials are constrained to be continuous, and to have continuous first and second derivatives. Bottom Right: A linear spline is shown, which is constrained to be continuous.

Regression splines

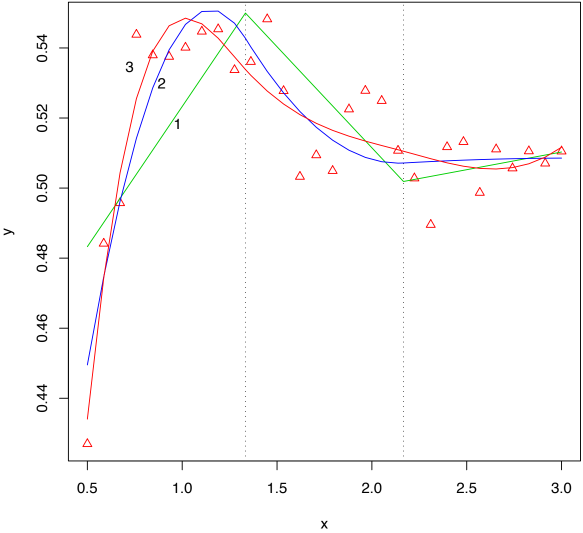
- We have n observed points (x_i, y_i) for $i = 1, \ldots, n$ that we want to interpolate
- we apply these ideas to parametric regression, by fitting a cubic spline ($d = 3$) to the n points
- we divide the x-axis into $K + 1$ intervals separated by K knots, $\xi_1, \ldots, \xi_K$, and interpolate the n points with the least squares criterion
- the obtained function is called regression spline

Regression splines

- The number K of knots and their position along the x-axis need to be chosen
- Because K is a tuning parameter regulating the complexity of the model, we need to perform a model selection according to bias-variance trade-off
- Once the number K has been set, a reasonable choice for knots position is uniformly along the x_i range.

Regression splines

Interpolated functions for $d = 1, 2, 3$



Smoothing splines

- Let us consider the penalized least squares criterion

$$D(f, \lambda) = \sum_{i=1}^{n} [y_i - f(x_i)]^2 + \lambda \int_{-\infty}^{\infty} \{f''(t)\}^2 \, \mathrm{d}t$$

where λ is a positive penalisation parameter of the roughness degree of curve f (quantified by the integral of $f''(x)^2$), and therefore acts as a smoothing parameter.

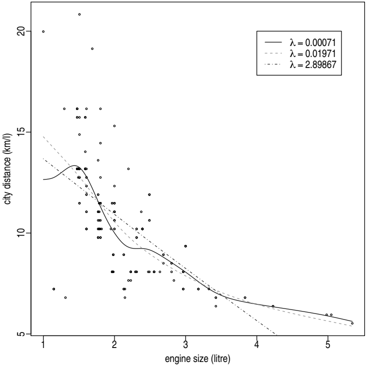
- Loss and Penalty formulation
- the term $\sum_{i=1}^{n} [y_i - f(x_i)]^2$ is a loss function that encourages $f(x)$ to fit the data well
- the term $\lambda \int_{-\infty}^{\infty} \{f''(t)\}^2 \, \mathrm{d}t$ is a penalty term that penalizes the variability in $f(x)$

Smoothing splines

- When $\lambda = 0$, then the penalty term has no effect, and so the function f will be very jumpy and will exactly interpolate the training observations.
- When $\lambda = \infty$, f will be perfectly smooth- it will just be a straight line that passes as closely as possible to the training points.
- In this case, f will be the linear least squares line, since the loss function amounts to minimizing the residual sum of squares.
- For an intermediate value of λ, f will approximate the training observations but will be somewhat smooth.
- So λ controls the bias-variance trade-off of the smoothing spline.

Smoothing splines

Estimate of `city distance` according to `engine size` by a smoothing spline, for three choices of λ



A noteworthy mathematical result shows that the solution to that minimization problem is represented by a natural cubic spline whose knots are distinct points x_i.

Summarizing. . .

We have relaxed the linearity assumption while still attempting to maintain as much interpretability as possible. To this end, we consider approaches such as splines and local regression.

- Regression splines involve dividing the range of X into K distinct regions. Within each region, a polynomial function is fit to the data. However, these polynomials are constrained so that they join smoothly at the region boundaries, or knots. Provided that the interval is divided into enough regions, this can produce an extremely flexible fit.
- Smoothing splines are similar to regression splines, but arise in a slightly different situation. Smoothing splines result from minimizing a residual sum of squares criterion subject to a smoothness penalty.
- Local regression is similar to splines, but differs in an important way. The regions are allowed to overlap, and indeed they do so in a very smooth way.
- Generalized additive models allow us to extend the methods above to deal with multiple predictors.

Generalized Additive Models

Generalized Additive Models

- So far we have seen a number of approaches for flexibly predicting a response y on the basis of a single predictor x. These approaches may be seen as extensions of simple linear regression.
- Here we explore the problem of flexibly predicting y on the basis of several predictors, $x_1, \ldots, x_p$.
- Generalized additive models (GAMs) provide a general framework for extending a standard linear model by allowing non-linear functions of each of the variables, while maintaining additivity.
- The beauty of GAMs is that we can use splines and local regression as building blocks for fitting an additive model

Generalized Additive Models

- A natural way of extending the multiple linear regression model

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i$$

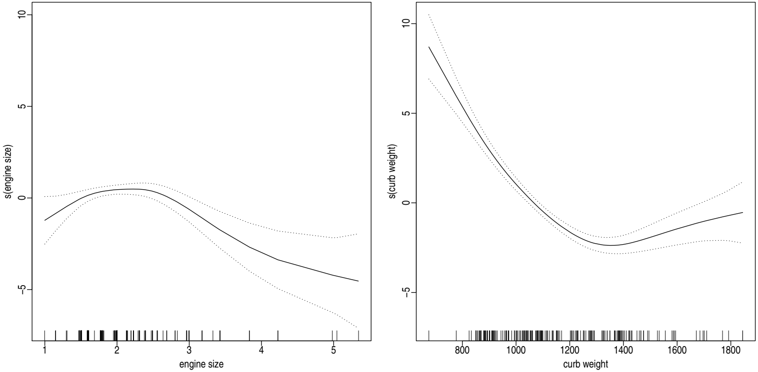
  in order to allow for nonlinear relationships between each feature and the response is to replace each linear component $\beta_j x_{ij}$ with a smooth nonlinear function $f_j(x_{ij})$.
- We can then write

$$y_i = \beta_0 + \sum_{i=j}^{p} f_j(x_{ij}) + \varepsilon_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}) + \varepsilon_i$$

- this is a Generalized Additive Model (GAM).
- It is called additive because we calculate a separate f_j for each x_j and then add together all of their contributions.

Example

Estimate of city distance according to engine size and curb weight by an additive model with a spline smoother



Generalized Additive Models

GAM important properties:

- GAMs allow us to fit a non-linear f_j to each x_j, so that we can automatically model non-linear relationships that standard linear regression will miss. This means that we do not need to manually try out many different transformations on each variable individually.
- The non-linear fits can potentially make more accurate predictions for the response y.
- Because the model is additive, we can still examine the effect of each x_j on y individually while holding all of the other variables fixed.
- If we are interested in inference, GAMs provide a useful representation.

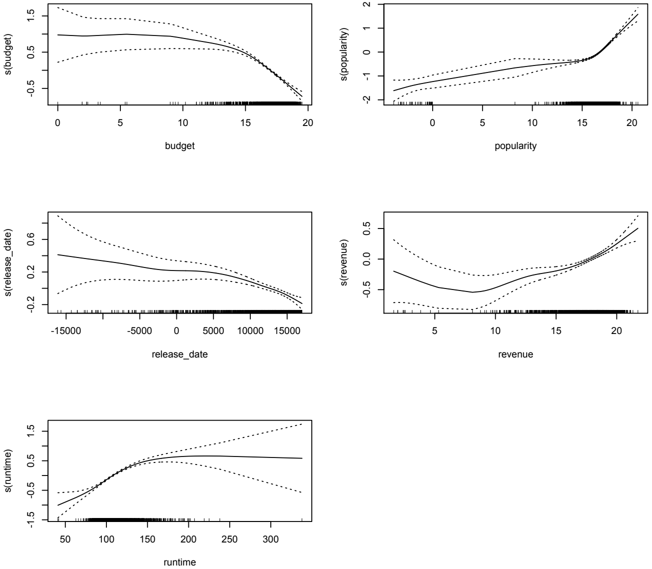
Example

Rating movies

- We consider the case of a dataset about movies
- We are interested in understanding the variable 'average vote' obtained by movies
- We want to study the relationship with other variables such as 'budget', 'popularity', 'revenues', 'runtime'

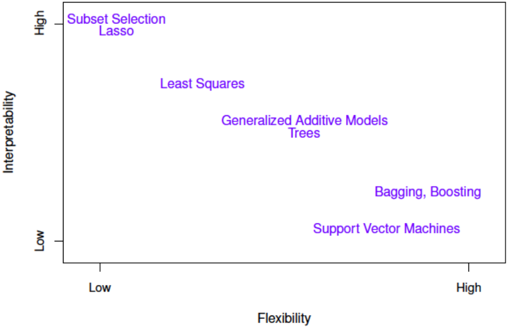
Generalized Additive Models

we may appreciate some results obtained with GAM



Fexibility vs Interpretability of models

There is a trade-off between flexibility and interpretability



Example

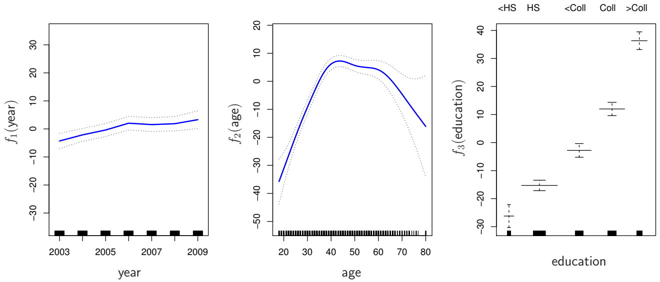
Let us consider the task of fitting the model

$$\text{wage} = \beta_0 + f_1(\text{year}) + f_2(\text{age}) + f_3(\text{education}) + \varepsilon$$

- year and age are quantitative variables
- education is qualitative with five levels:
  <HS, HS, <Coll, Coll, >Coll, referring to the amount of high school or college education that an individual has completed.
- we fit a model where f_1 and f_2 are smoothing splines

Example

Wage data



- holding age and education fixed, wage tends to increase slightly with year
- holding education and year fixed, wage tends to be highest for intermediate values of age, and lowest for the very young and very old.
- holding year and age fixed, wage tends to increase with education: the more educated a person is, the higher their salary, on average.

Example

Wage data: GAM with smoothing splines for year and age

```
Anova for Parametric Effects
              Df  Sum Sq Mean Sq F value    Pr(>F)
s(year, 4)     1   27162   27162  21.981 2.877e-06 ***
s(age, 5)      1  195338  195338 158.081 < 2.2e-16 ***
education      4 1069726  267432 216.423 < 2.2e-16 ***
Residuals   2986 3689770    1236
---

Anova for Nonparametric Effects
            Npar Df Npar F  Pr(F)
(Intercept)
s(year, 4)        3  1.086 0.3537
s(age, 5)         4 32.380 <2e-16 ***
education
```

$$AIC = 29887.75$$

Example

Wage data: GAM with smoothing spline for age

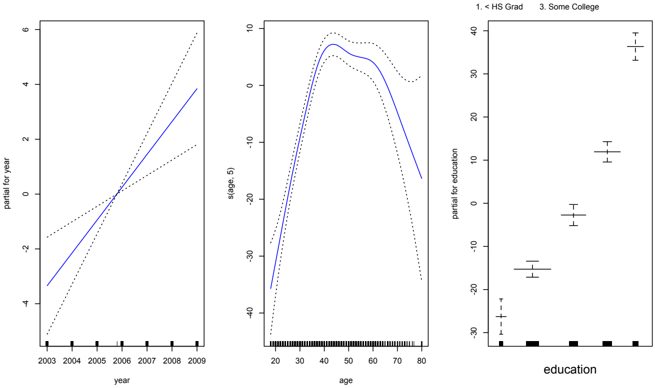
```
Anova for Parametric Effects
              Df  Sum Sq Mean Sq F value    Pr(>F)
year           1   27154   27154  21.973  2.89e-06 ***
s(age, 5)      1  194535  194535 157.415 < 2.2e-16 ***
education      4 1069081  267270 216.271 < 2.2e-16 ***
Residuals 2989 3693842    1236
---


Anova for Nonparametric Effects
              Npar Df Npar F     Pr(F)
(Intercept)
year
s(age, 5)           4  32.46 < 2.2e-16 ***
education
```

$AIC = 29885.06$

Example

Wage data



Example

Wage data: GAM with smoothing spline for year and local regression for age

```
Anova for Parametric Effects
                           Df  Sum Sq Mean Sq F value    Pr(>F)
s(year, df = 4)           1.0   25188   25188  20.255 7.037e-06 ***
lo(age, span = 0.7)       1.0  195537  195537 157.243 < 2.2e-16 ***
education                 4.0 1101825  275456 221.511 < 2.2e-16 ***
Residuals              2988.8 3716672    1244
---

Anova for Nonparametric Effects
                    Npar Df Npar F  Pr(F)
(Intercept)
s(year, df = 4)         3.0  1.103 0.3464
lo(age, span = 0.7)     1.2 88.835 <2e-16 ***
education
```

$$AIC = 29903.95$$

Example

Wage data: linear model

```
Anova for Parametric Effects
            Df  Sum Sq Mean Sq F value    Pr(>F)
year         1   22434   22434  17.421  3.08e-05 ***
age          1  195045  195045 151.460 < 2.2e-16 ***
education    4 1150320  287580 223.317 < 2.2e-16 ***
Residuals 2993 3854286    1288
```

$$AIC = 30004.62$$