Statistical Methods - Nonparametric Regression

Lecture 5

Mattias Villani

Sveriges Riksbank and Stockholm University

April 28, 2010

Overview Lecture 5

- Linear regression as linear smoother
- Transformations
- Polynomial regression - a global smoother
- Nearest neighboor and Kernel regression
- Regression trees

- The linear regression model in vector/matrix form

$$\underset{(n\times 1)}{y} = \underset{(n\times p)(p\times 1)}{X\beta} + \underset{(n\times 1)}{\varepsilon}$$

$$y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}, \ \beta = \begin{pmatrix} \beta_1 \\ \vdots \\ \beta_p \end{pmatrix}, \ \varepsilon = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

$$X = \begin{pmatrix} x_1' \\ \vdots \\ x_n' \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix}$$

- Usually $x_{i1} = 1$, for all i. β_1 becomes the intercept.
- $\varepsilon \sim N(0, \sigma^2 I_n)$

- OLS/MLE of β
$$\hat{\beta} = (X'X)^{-1}X'y$$
- Unbiased estimator of σ^2 (not MLE)
$$s^2 = \frac{(y - X\hat{\beta})'(y - X\hat{\beta})}{n-p}$$
- Covariance matrix of $\hat{\beta}$
$$Cov(\hat{\beta}) = \sigma^2(X'X)^{-1}.$$
- Sampling distribution of $\hat{\beta}$
$$\hat{\beta} \sim t_{n-p}\left[\beta, s^2(X'X)^{-1}\right]$$
- The same results are obtained from Bayesian analysis with the non-informative prior
$$p(\beta, \sigma^2) \propto \sigma^{-2}$$

- The predicted values
$$\hat{y} = X\hat{\beta} = X(X'X)^{-1}X'y = Hy$$
where
$$H = X(X'X)^{-1}X'$$
is the important **hat matrix**.
- The relation $\hat{y} = Hy$ shows that each fitted value is a linear combination of the y-values in the sample, i.e.
$$\hat{y}_i = \sum_{j=1}^{n} H_{ij} y_j.$$
- The linear regression model with the MLE is a linear smoother.
- Any regression model that generates a prediction rule of the form
$$\hat{y} = Ly$$
is called a **linear smoother**. Many non-parametric regression models are linear smoothers, e.g. polynomial and kernel regression, and splines.

The hat matrix and residuals

- The residual vector can be written

$$e = y - \hat{y} = y - Hy = (I_n - H)y$$

which has the implication

$$Cov(e) = (I - H)Cov(y)(I - H)' = \sigma^2(I - H).$$

This means that the hat matrix is also very important in residual testing, since e.g.

$$St.dev.(e_i) = \sigma\sqrt{(1 - H_{ii})}$$

- This is used when defining studentized residuals and Cook's distance.

Properties of the hat matrix

- The hat matrix is symmetric and idempotent (equals its own powers)

$$H = H' = H^2.$$

- The relation $\hat{y}_i = \sum_{j=1}^{n} H_{ij} y_j$ shows that H_{ij} is a measure of the **influence** that observation j has on the fit of observation i. Since H depends only on X and not on y, the H_{ij} are measures of *potential* influence.
- The trace of the hat matrix measures the degrees of freedom in the fitting:

$$tr(H) = tr\left[X(X'X)^{-1}X'\right] = tr\left[(X'X)(X'X)^{-1}\right] = tr(I_p) = p.$$

  [here we have used the general result $tr(AB) = tr(BA)$].
- It turns out that $tr(L)$ is a generalization of the degrees of freedom concept to the class of linear smoothers.

- When the data are non-linear (or non-normal) we can always try to transform the data to linearity.
- Box-Cox transformation

$$y(\lambda) = \begin{cases} \frac{y^{\lambda}-1}{\lambda} & \text{if } \lambda \neq 0 \\ \ln y & \text{if } \lambda = 0 \end{cases}$$

- Extended Box-Cox (can also handle negative value of y)

$$y(\lambda) = \begin{cases} \frac{(y^{\lambda_1}+\lambda_2)-1}{\lambda_1} & \text{if } \lambda_1 \neq 0 \\ \ln(y+\lambda_2) & \text{if } \lambda_1 \neq 0 \end{cases}$$

  where λ_2 is set so that $y + \lambda_2 > 0$ for all relevant y.
- We can estimate λ jointly with β in the model

$$y(\lambda) = x'\beta + \varepsilon.$$

- But transforming the response y to obtain linearity may mess up higher order moments. Also, in some cases it is hard/impossible to obtain linearity, see the Lidar example in RCW.

- Polynomial of order k. Original covariate, x. Extended set of covariates:

$$z' = (z_0, z_1, z_2, ..., z_k) = (1, x, x^2, ..., x^k)$$

- Basis functions. Basis expansion.
- The polynomial regression is obviously non-linear in (y, x) space when $k > 1$:

$$E(y|x) = \beta_0 + \beta_1 x + \beta_2 x^2 + ... + \beta_k x^k$$

- But the polynomial regression is linear with respect to (y, z) space:

$$E(y|z) = \beta_0 + \beta_1 z_1 + \beta_2 z_2 + ... + \beta_k z_k = z'\beta$$

and can therefore be fitted by OLS/MLE in the basis space

$$\hat{\beta} = (Z'Z)^{-1}Z'y.$$

Polynomial regression is a global linear smoother

- The fitted values of a polynomial regression are

$$\hat{y} = Ly,$$

  where $L = Z(Z'Z)^{-1}Z'$. Hence polynomial regression is a linear smoother.
- The degrees of freedom are $tr(L) = k + 1$.
- **Local smoother**: Changing an observation y_i only affects the fit of other observations that have are close in covariate space to y_i.
- The polynomial regression is a **global smoother**. Changing an observation y_i will typically affect the fit at other distant observations.
- Graphical display of smoother matrix is helpful.

Nearest Neighbor regression

- The **k-nearest-neighbor estimator** takes the local aspect seriously:

$$\hat{y} = \hat{f}(x) = Ave\left[y_j | x_j \in \mathcal{N}_k(x)\right],$$

where $Ave()$ denotes the average and $\mathcal{N}_k(x)$ denotes the neighborhood in covariate space that contains exactly the k nearest neighbors to x.

- The **k-nearest-neighbor** fit of an observation is therefore just the average of its k nearest neighbors (in covariate space).
- Large variance, low bias. General point: Bias-Variance trade-off.
- Discontinuous.

- Obvious way of getting continuity: Take a local weighted average with weights that die of as we move away from x:

$$\hat{f}(x) = \frac{\sum_{i=1}^{n} K\left(\frac{x_i - x}{b}\right) y_i}{\sum_{i=1}^{n} K\left(\frac{x_i - x}{b}\right)},$$

where $K\left(\frac{x_i - x}{b}\right)$ is the **Kernel weighting function** with **bandwidth** $b > 0$.

Examples of kernel functions

- Epanechnikov

$$K\left(\frac{x_i - x}{b}\right) = \frac{3}{4}\left[1 - \left(\frac{x_i - x}{b}\right)^2\right] \text{ if } \left|\frac{x_i - x}{b}\right| \leq 1,$$

  and zero otherwise.

- Uniform

$$K\left(\frac{x_i - x}{b}\right) = 1 \text{ if } \left|\frac{x_i - x}{b}\right| \leq 1,$$

  and zero otherwise.

- Gaussian

$$K\left(\frac{x_i - x}{b}\right) = \phi\left(\frac{x_i - x}{b}\right).$$

- The tricky part is to choose the bandwidth, b.

Local polynomial regression

- Generalizes the Nadaraya-Watson estimator by fitting a polynomial locally at each x:

$$y_i = \beta_0^{(x)} + \beta_1^{(x)}(x_i - x) + ... + \beta_k^{(x)}(x_i - x)^k + \varepsilon_i,$$

using weighted least squares with weights

$$w_i^{(x)} = K\left(\frac{x_i - x}{b}\right).$$

- Note that the regression coefficients change from x to x, we are fitting a bunch of regressions, one at each x of interest.
- All the local regression fit use all n observations, but gives more weight to observations with covariates that are closer to x.
- By construction, the fit at x is $\hat{f}(x) = \beta_0^{(x)}$.
- Local poynomial regression is a linear smoother.
- Problem: Hard to generalize to situations with many covariates.

Regression trees

- Regression trees divides the covariate space into regions and then fits a constant within each region.
- The regions $R_1, ..., R_Q$ are constructed from a sequence of binary splits, e.g.
  - $x_{i_1} \leq l_1$, $i_1 \in \{1, 2, ..., k\}$
  - $x_{i_2} \leq l_2$, $i_2 \in \{1, 2, ..., k\}$
  - ...
- A regression tree is therefore of the form

$$\hat{f}(x) = \sum_{q=1}^{Q} c_q I(x \in R_q)$$

- We need to estimate:
  - The number of splits, Q
  - The regions, $R_1, ..., R_Q$, or equivalently, the sequence of splitting variables $i_1, i_2, ...$ and the split points, $l_1, l_2, ...$
  - The constants in each region $c_1, c_2, ...c_Q$.

Regression tree example: cars milage

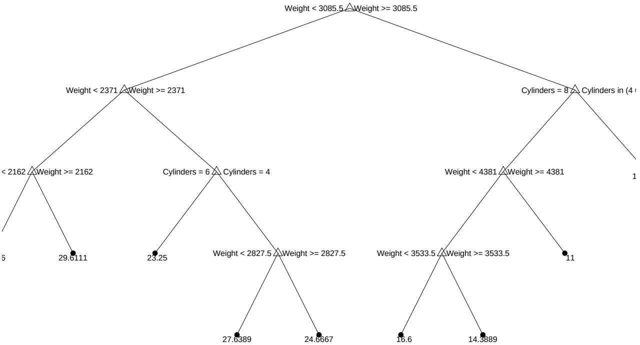

Figure: Graphical representation of the regression tree for the milage data.