Lecture 9

Casey Gibson

October 2017

Smoothing (Agresti. 7.4)

- Kernel Smoothing
- Penalized likelihood
- GAMS

Bias-Variance trade-off

Fundamental question about choice of statistical models

- Parametric models:
  - Represent data 'parsimoniously' (with few parameters)
  - good for variability of estimates
  - Bad if model is wrong
  - 'bad for bias, good for variance'
- Less Parametric models (e.g. smoothers)
  - non-parametric
  - semi-parametric
- less likely to be biased b/c less structure
- more uncertainty/variability in estimates

Example 3.3.8

- Illustrates a situation where even when the model is wrong the MSE is lower for the parametric model than a non-parametric model
- Takeaway: a little miss-specification can be okay, but a lot is bad

Smoothing

- Why smooth?
  - Exploratory analysis
  - more concerned about bias than variance (large N)
  - simple way to 'gently' relax the assumptions of your model
- Main examples
  - Kernel Smoothing
  - Penalized likelihood
  - GAMS

Kernel Smoothing

- Contingency Table
  - Joint cell probabilities $\vec{\pi}$
  - sample probabilities $\vec{p}$
  - Suppose we are interested in $\tilde{\pi}$, smoothed estimates of $\vec{\pi}$
  - We can do this using a smoothing matrix K where $\tilde{\pi} = Kp$, square, non-negative elements, column totals sum to $1 \rightarrow \sum_j \pi_j = 1$

Specifically

$$\tilde{\pi_i} = (1 - \lambda)p_i + \lambda * [smoother_i]$$

where $smoother_i$ is defined by K.
For example, it could be average of near-by cells.
We can see that if $\lambda = 0$

$$\tilde{\pi_i} = p_i$$

or if $\lambda = .5$

$$\tilde{\pi_i} = p_i + \frac{smoother_i}{2}$$

- Regression context
  - Smoothing across an X
  - $\tilde{\pi(x)} = \frac{\sum_i y_i \phi(\frac{x - x_i}{\lambda})}{\sum_i \phi(\frac{x - x_i}{\lambda})}$
  - ϕ is a symmetric kernel function
    - $\int_{-\infty}^{\infty} K(u)du = 1$
    - $K(-u) = K(u) \forall u$
    - Example: Gaussian kernel $K(u) = e^{\frac{1}{2\sigma^2}(u-\mu)^2}$

- $\lambda > 0$ is smoothing parameter
- **Takewaway**: Take a weighted sum of all of my ys where the weight is proportional to the similarity of the test x to the xs in the data

Example 1
If $\phi(u) = 1if u = 0, 0otherwise$
If

$$\begin{cases} 1 & u = 0 \\ 0 & otherwise \end{cases}$$

then

$$\tilde{\pi(x_k)} = \text{sample proportion of successes when } X = X_k$$

Example 2

$$\phi(i) = exp(\frac{-u^2}{2})$$

Equivalent to $N(0, 1)$

$$\tilde{\pi(x)} = \sum_i y_i exp(-\frac{1}{2}\frac{(x - x_i)^2}{\lambda})$$

As $\lambda \to 0$ we approach no smoothing which implies low bias $\pi(x)$. We only have a few data points being used to estimate in a neighborhood around x.

Penalized Likelihood (7.4.5)

Is a

- 'regularization' method
- 'smoothing' method 'penalization' method
- 'shrinkage' method

$$L^*(\beta) = L(\beta) - \lambda(\beta)$$

Where $\lambda(\beta) = p(\beta, \lambda)$
Very general and broad approach for giving sensible estimates for parameters/characteristics that are unstable.

Commonly used penalizations

$$L_2 - norm\ penalty : \lambda(\beta) = \lambda \sum_j \beta_j^2$$

$$L_1 - norm\ penalty : \lambda(\beta) = \lambda \sum_j |\beta_j|$$

Similar to maximizing likelihood subject to the constraint that

$$\sum_j |\beta_j| \leq K$$

$$L_0 - norm\ penalty : \lambda(\beta) = K \sum_j \mathbf{1}(|\beta_j| > 0)$$

penalty proportional to how many are not 0.
AIC and BIC are special cases of this.
Difficult to optimize for large J.
How to choose λ?

- K-fold cross validation
  - fix λ
  - leave out $\frac{1}{kth}$ of dataset
  - Fit your model, predict for the left out data
  - repeat steps 1-2 until we have out of sample predictions for the whole dataset
  - Compute some error for λ
  - Repeat for new λs
  - Choose λ_i that minimizes error

Penalized likelihood $\approx$ Bayes
If our prior for $\beta \propto e^{-\lambda(\beta)}$ then $posertior(\beta) \propto L^*(\beta)$ so posterior mode = mdoe of $L^*(\beta)$

Generalized Additive Models

$$g(\mu_i)$$

where g is a link and μ_i is a mean

$$g(\mu_i) = \sum_j s_j(x_{ij})$$

where s is a smooth function, i.e. a cubic spline

- GLMS are a special case when s_j are linear.

    - Models are fit using a "back fitting algorithm"
    - Similar to Newton-Raphson w/ local smoothing
    - Penalized likelihood that penalizes "wiggliness" of function.
    - Approximate d.f. for each
- These models are more flexible than GLMs but are also much less interpret-able.
- Note: In *R* the *mgcv* package is able to estimate GLMs