28

OA - MARS

Multivariate Adaptive Regression Splines

- Form of regression analysis
- Non parametric - model no assumptions about the underlying functional relationship between the dependent and independent variables
- Fits piecewise linear regressions.
- Use of separate regression slopes in distinct intervals of the independent variable space
- Allows look for interaction between variables

- Construct a set of basis functions β_i in a certain manner

$$\hat{f} = \sum_{m=1}^{M} a_m \beta_m(x)$$

- An extension of stepwise linear regression
- Modification of CART to improve its performance in regression setting

- MARS build models in the form of $\hat{f}(x) = \sum_{i=1}^{k} c_i B_i(x)$
- c_i are coefficients
- $B_i(x)$ are called basis functions

Basis function

- Piecewise linear basis function sometimes called hinge functions
- $(x-t)_+$ and $(t-x)_+$
- The + means the positive part.

$$(x-t)_+ = \begin{cases} x-t & \text{if } x > t \\ 0 & \text{otherwise} \end{cases}$$

$$(t-x)_+ = \begin{cases} t-x & \text{if } x < t \\ 0 & \text{otherwise} \end{cases}$$

- If we have N cases and p variables with all cases having distinct values for each variable, there are 2NP possible basis functions

- Can take 1 of 3 forms - a basis function -
  - a constant
  - a hinge function $(x-t)_+$ or $(t-x)_+$
  - Product of hinge functions.

Reflected Pairs
- Reflected pairs at a particular knot value
- For each input x_i, calculate reflected pairs with knots at each individual value of x_i
- Collection of basis functions C
- $C = [(x_j - t)_+, (t - x_j)_+]$
  $t \in [x_{1j}, x_{2j}, \ldots x_{Nj}]$ individual unique value for each variable
  $j = 1, 2, \ldots p$ variables

- Stepwise linear regression
- Use the reflective pairs or product of reflective pairs
  $f(x) = \beta_0 + \sum_{m=1}^{n} \beta_m h_m(x)$
- $h_m(x)$ is a function in C - pair of reflective functions
  Or a product of two or more such functions.
  β_m estimated by minimizing the residual sum-of-squares

Procedure
- We start with constant function $h_0(x) = 1$
- Define set of terms in model as ϕ
- All products of a function h_m in the model set ϕ with one of the reflected pairs in C.
- We add this to the model if a term of the form
  $\beta_{m+1} h_l(x) * (x_j - t)_+ + \beta_{m+2} h_l(x) * (t - x_j)_+ \quad h_l \in \phi$
- Calculate weights
- We add the term that produce the largest decrease in the training error - residual sum of squares

DA - MARS

Example:
- First term is a constant h_0
- Second term
- We now consider a function of the form $\beta_1(x_j - t)_+ + \beta_2(t - x_j)_+$
- For example, Suppose ~~on~~ turns out to be $\beta_1(x_7 - t)_+ + \beta_2(t - x_7)$ for some value of t
- Multiplication by a constant does not change things.

- next Stage we add $h_m(x) * (x_j - t)_+$ and $h_m(x) * (t - x_j)_+$
- We have 3 choices for h_m:
  $h_0(x) = 1$ i.e. constant.
  $h_1(x) = (x_7 - t)_+$
  $h_2(x) = (t - x_7)_+$
- The last two are interaction terms.
- At end of process we have a very large model
- Set a limit on the number of terms in model

Hierarchical Structure
- Two way interactions included if main effects are there.
- A 4-way product only included if one of its three way components in the model already
- Can restrict interaction
- Each input can appear at most once in a product.

Forward Pass
- Add terms in pairs until you:
  - Reach max # of terms
  - Adding a term changes R^2 by less than 0.001
  - Reached an R^2 of 0.999 or more
  - GRSq < -10
  - No new term increases R^2

- Default for n_m is $\min(200, \max(20, 2 * ncol(x))) + 1$

Backward Pass
- Need to prune the number of Basis functions
- Can specify max # of terms have - nprune
- Assume that we have n_b basis functions
- For each subset $1 \ldots n_b$ Find best subset in terms of lowest RSS
- Then look at each of these subsets and calculate GCV to find lowest value
- This gives us set of basis functions to use
- Calculate the coefficients, residuals and fitted values by lm

- Typically overfits.
- Go backwards and remove and remove the terms which cause the smallest decrease in the residual sum of squared
- Estimate model for each size λ
- Use cross validation to estimate λ

General Cross Validation formula

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

- $M(\lambda)$ is the effective number of parameters in model
- r linearly independent basis functions
- k knots
- $M(\lambda) = r + 3k$

$$GRSq = 1 - \frac{GCV}{GCV.null}$$

- where GCV.null is the GCV of an intercept only model
- The GCV and GRSq are measures of the generalization ability of the model ie how well the model would predict using data not in the training set.

28/04/16

DA - MARS

OUTPUT PLOTS

1 - Model Selection
- Plot of R^2 and $GRSq$
- R-Sqe - a normalised version of the RSq
- $GRSq$ - measure of generalisation ability of the model
- Should not run together - should be an increased penalty being applied to the GCV as # of model parameters increases
- GCV - Generalised Cross Validation : trade off goodness of fit v model complexity

2 - Residuals vs Fitted
- Shows residuals of each value of predicted response : (remainder of observed - expected value)
- Should be as close to zero as possible
- Constant variance of residuals not as important as in linear models
- Highlights outliers

3 - Cumulative Distribution
- Cumulative Sums of absolute values of residuals
- Ideally, Start at zero and Shot Straight up to one
- Can calculate mean proportion 50% residual value at which 95% of values are contained within

4 - Quantile Quantile
- Compares the distribution of residuals to the assumed normal distribution
- Want all points to fall on a straight line
- Look for outliers

Variable Importance
- Look relationship between var importance in model compared to data
- Variance of variable importance is high
- Run different datasets → bootstrap → may give different answers
- Highly correlated variables - one variable chosen over the other

- In interaction, each variable gets credit for entire term
- Use 3 different criteria:
  - # times variable appears in subset of various sizes
  - Decrease in RSS for each subset relative to previous subsets.
  - Decrease in GCV for each subset relative to previous subset
  Normalized so that the larger decrease is 100

i.e. variable with "highest importance" will have large number for number and number close to 100 for GCV and RSS

Comments
- Can calculate prediction and CI.
- Develop a variance model
- Assumes errors are independent
- Ability to operate locally
- Regression surface built up parsimoniously using non-zero components only where they are needed
- Can use GLM method after the base model has been constructed to determine final weights

- Can be computationally slow to fit the model and for more complex than a tree
- Has advantage of being able to be used on binary or continuous outcome
- Non parametric regression procedure - makes no assumption about underlying functional relationship between dependent and independent variables
- Useful for and identifies interactions
- Typically useful
- Automatically models non-linearities and interactions

28/04/16 DA-MARS 7.

- More flexible than linear regression models
- Simple to understand
- Often requires little or no data preparation - effect of outliers is contained
- Automatic variable selection
- Tend to have a good bias-variance trade off ⇒ model is flexible enough to model non-linearity and variable interactions (low bias) yet the constrained form of MARS basis functions prevents too much flexibility (low variance)
- Suitable for handling large datasets
- Cross validation and related techniques may be used for validating the model
- Doesn't give as good a fit as boosted trees, but can be built quicker and are more interpretable

- Can enter linear pred - variables that are tell important and flexibility to added in as consideration for a hinge function - reduces complexity, fit is to see

- Cross validation - partition data into k-fold subsets, repeatedly build model on all but one of those subsets, measure performance on the left out data