Statistical Learning for Analyzing Functional Genomic Data

Axel Benner

German Cancer Research Center, Heidelberg, Germany

June 16, 2006

Genomics, Transcriptomics and Proteomics in Clinical Research

- Diagnostics
  - signatures
  - single biomarkers
- Prognostic Factor Studies
  - response to treatment
  - toxicity
  - survival
- Custom Drug Selection
  - predictive factors for response/ resistance to certain therapy
  - indicators of adverse events
- Discovery of Therapeutic Targets
  - candidate targets
- Insight in Pharmacological Mechanisms
  - pathway analysis

Explanation vs. Prediction

- Target: Explanation
  - Implies that there is some likelihood of a "true" model
  - Model selection: few input variables are relevant
  - Occam's razor: 'do not make more assumptions than needed'
- Target: Prediction
  - Statistical learning
  - Model selection: quality of prediction
- Topic: Large scale problems

Large scale problems

- New biomolecular techniques:
  - Number of input variables (genes, clones, etc.): 1000s to 10,000s
  - Number of observations: 10s to 100s
    $\rightarrow$ number of observations $<<$ number of input variables
    $\rightarrow$ more unknown parameters than estimation equations
    $\rightarrow$ infinitely many solutions
- Models can be fit perfectly to the data
  $\rightarrow$ no bias but high variance
- Use statistical learning methods to handle these problems!

Statistical Learning

Control of Model Complexity

- Restriction methods
  - the class of functions of the input vectors is limited
- Selection methods
  - constitute methods, which include only those basis functions of the input vectors that contribute 'significantly' to the fit of the model
  - examples are variable selection methods, stepwise greedy approaches like boosting
- Regularization methods
  - restrict the coefficients of the model, e.g. ridge regression

Penalized maximum likelihood estimation

- Maximizing the log likelihood can result in fitting noise in the data.
- A shrinkage approach will often result in estimates of the regression coefficients that, while biased, are lower in mean squared error and are more close to the true parameters.
- A good approach to shrinkage is penalized maximum likelihood estimation (le Cessie & van Houwelingen, 1990).
- A general form of penalized log likelihood is

$$\sum_{i=1}^{n} logL(y_i; g(x_i^T \beta)) - \sum_{j=1}^{d} p_\lambda(|\beta_j|)$$

From the log-likelihood a so-called 'penalty' is subtracted, that discourages regression coefficients to become large.

Penalty functions

A good penalty function should result in a estimator with the following three properties (Fan & Li, 2001):

- Unbiasedness: The resulting estimator is nearly unbiased when the true unknown parameter is large to avoid excessive estimation bias
- Sparsity: Estimating a small coefficient as zero, to reduce model complexity
- Continuity: The resulting estimator is continuous in the data to avoid instability in model prediction

Penalty functions

Well-known penalty functions are L_q-norm penalties:

$$p_\lambda(|\theta|) = \lambda|\theta|^q$$

- L_2 (Ridge regression) with thresholding rule

$$\hat{\theta}(z) = \frac{1}{1+\lambda} z$$

$\rightarrow$ continuous, but biased and no sparse solutions

- L_1 (LASSO) with thresholding rule

$$\hat{\theta}(z) = sgn(z)(|z| - \lambda)_+$$

$\rightarrow$ continuous and sparse, but no unbiased solutions

Penalty functions

- Convex penalties (e.g. quadratic penalties)
  - make trade-offs between bias and variance
  - can create unnecessary biases when the true parameters are large
  - parsimonious models cannot be produced
- Nonconcave penalities
  - select variables and estimate coefficients of variables simultaneously
  - e.g. hard thresholding penalty (HARD, Antoniadis 1997)

$$p_\lambda(|\theta|) = \lambda^2 - (|\theta| - \lambda)^2 I(|\theta| < \lambda)$$

with thresholding rule

$$\hat{\theta} \;=\; z \cdot I(|z| > \lambda)$$

Penalty functions

Related approaches

- Bridge regression (Frank & Friedman, 1993) which minimizes $\sum(y_i - \beta_0 - \sum_j \beta_j x_{ij})^2$ subject to $\sum_{j=1}^{d} |\beta_j|^\gamma \leq t$ with $\gamma \geq 0$.
- Nonnegative garotte (Breiman, 1995), which minimizes $\sum(y_i - \beta_0 - \sum_j c_j \beta_j x_{ij})^2$ under the constraint $\sum c_j \leq s$ where $\{\hat{\beta}_j\}$ are the full-model OLS coefficients.
- Elastic net (Zou & Hastie, 2005), where the penalty is a convex combination of the lasso and ridge penalty.
- Relaxed Lasso (Meinshausen, 2005).

SCAD penalty

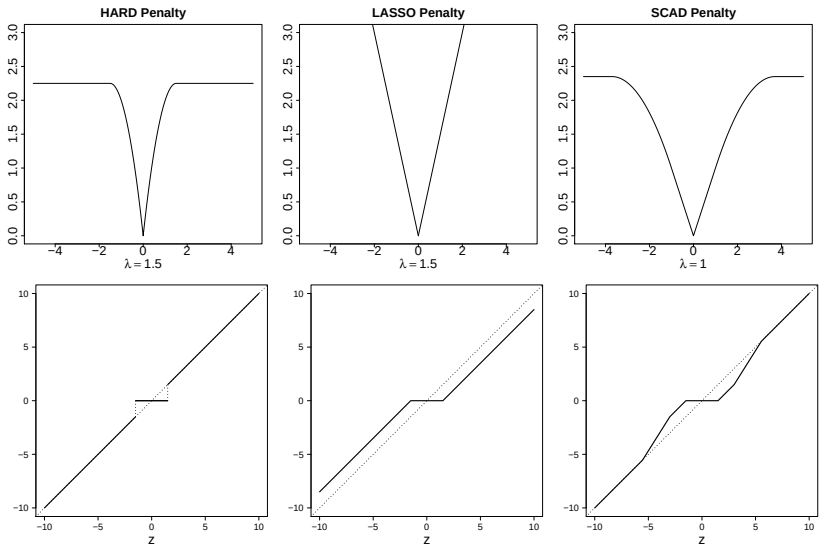
- Smoothly Clipped Absolute Deviation (SCAD; Fan, 1997)
  - satisfies all three requirements (unbiasedness, sparsity, continuity)
  - is defined by

$$p'_\lambda(|\theta|) \;=\; \lambda \left\{ I(|\theta| \leq \lambda) + \frac{(a\lambda - |\theta|)_+}{(a-1)\lambda} I(|\theta| > \lambda) \right\}, \quad a > 2$$

with thresholding rule

$$\hat{\theta}(z) \;=\; \begin{cases} sgn(z)(|z| - \lambda)_+, & |z| \leq 2\lambda \\ \{(a-1)z - sgn(z)a\lambda\}/(a-2), & 2\lambda < |z| \leq a\lambda \\ z, & |z| > a\lambda \end{cases}$$

Selected penalty and thresholding functions



SCAD Penalty

- SCAD improves the LASSO via reducing estimation bias.
- SCAD possesses an oracle property:
  the true regression coefficients that are zero are automatically estimated as zero, and the remaining coefficients are estimated as well as if the correct submodel were known in advance.
- Hence, SCAD is an ideal procedure for variable selection, at least from theoretical point of view.

Penalized proportional hazards regression

Penalized partial likelihood

$$l(\boldsymbol{\beta}) - \sum_{j=1}^{d} p_\lambda(|\beta_j|) \rightarrow \max_{\boldsymbol{\beta}}$$

with

$$l(\boldsymbol{\beta}) = \sum_{k=1}^{N} [\mathbf{x}_{(k)}^T \boldsymbol{\beta} - \log\{\sum_{i \in R_k} \exp(\mathbf{x}_i^T \boldsymbol{\beta})\}].$$

where n = number of observations,
N = number of events,
R_k = risk set for event k, $k = 1, ..., N$.

SCAD Regression

SCAD Regression (Fan & Li, 2002)

- Use 'LQA', local quadratic approximation for β close to β_0,

  $$l(\beta_0) + \nabla l(\beta_0)^T(\beta - \beta_0) + \tfrac{1}{2}(\beta - \beta_0)^T \nabla^2 l(\beta_0)(\beta - \beta_0) - n\tfrac{1}{2}\beta^T \Sigma_\lambda(\beta_0)\beta$$

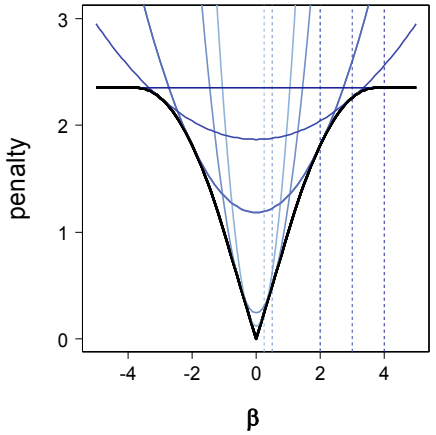
  with $\Sigma_\lambda(\beta_0) = diag\,\{p'_\lambda(|\beta_{10}|)/|\beta_{10}|, ..., p'_\lambda(|\beta_{d0}|)/|\beta_{d0}|\}$
- Solve quadratic maximization problem by Newton-Raphson algorithm

  $$\beta_1 = \beta_0 - [\nabla^2 l(\beta_0) - n\Sigma_\lambda(\beta_0)]^{-1}[\nabla l(\beta_0) - n\Sigma_\lambda(\beta_0)\beta_0]$$
- Estimate covariance matrix by sandwich formula

  $$cov(\hat{\beta}_1) = [\nabla^2 l(\hat{\beta}_1) - n\Sigma_\lambda(\hat{\beta}_1)]^{-1} cov(\nabla l(\hat{\beta}_1))[\nabla^2 l(\hat{\beta}_1) - n\Sigma_\lambda(\hat{\beta}_1)]^{-1}$$

SCAD Regression: Local quadratic approximation for $p_\lambda(\beta)$

Fan & Li, 2002



$$p_\lambda(|\beta_j|) \approx p_\lambda(|\beta_{j0}|) + 1/2\,\{p'_\lambda(|\beta_{j0}|)/|\beta_{j0}|\}\,(\beta_j^2 - \beta_{j0}^2) \quad \text{for } \beta_j \approx \beta_{j0}$$

SCAD Regression for $d > n$

1. Variable Reduction
   - Since $d > n$, we use the Singular Value Decomposition of $(n \times d)$-design matrix X (Hastie & Tibshirani, 2004):

$$X = USV^T = RV^T$$

   - With parameter transformation $\theta = V^T\beta$ perform a single step of SCAD estimation for θ and transform back to obtain $\hat{\beta}_0 = V\hat{\theta}$.
2. Variable Selection
   Perform SCAD regression (Fan & Li, 2002) with initial estimates from single step SCAD estimation, and start with

$$\hat{\beta}_{j0} = \begin{cases} \hat{\beta}_{j0} & |\hat{\beta}_{j0}| \geq c \cdot se(\hat{\beta}_{j0}) \\ 0 & |\hat{\beta}_{j0}| < c \cdot se(\hat{\beta}_{j0}) \end{cases}, \; j = 1, ..., d$$

$$\text{increase } c \text{ until } |\left\{\hat{\beta}_{j0} : \hat{\beta}_{j0} \neq 0\right\}| \leq n$$

(Hastie & Tibshirani(2004) Efficient quadratic regularization for expression arrays. Biostatistics)

Penalized Regression: Selecting penalty parameter λ

Selection of thresholding parameter

Estimate λ by minimizing an approximate generalized cross-validation (GCV) statistic (Craven & Wahba, 1977) regarding the penalized likelihood as an iteratively reweighted least-squares problem

$$GCV(\lambda) = \frac{-l(\hat{\beta})}{n[1 - e(\lambda)/n]^2}$$

where

$$e(\lambda) = tr[(\nabla^2 l(\hat{\beta}) - \Sigma_\lambda(\hat{\beta}))^{-1}\nabla^2 l(\hat{\beta})]$$

computes the effective degrees of freedom (d.f.) for this problem.

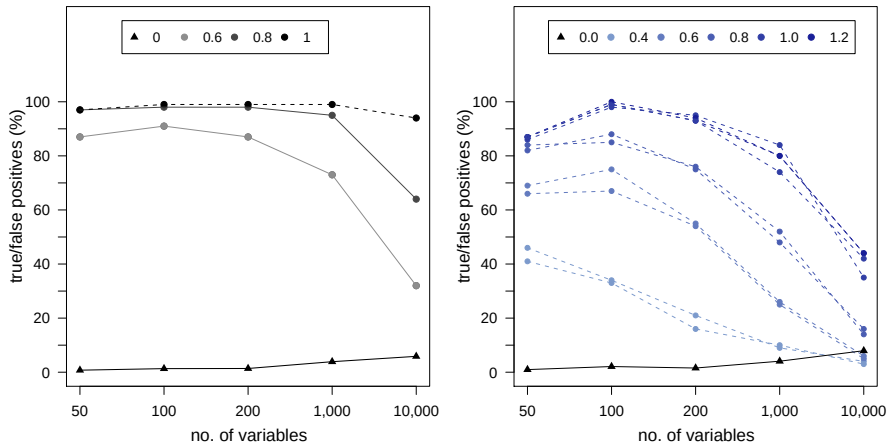
Simulation study

Artificial data (100 cases with $\approx$ 30 % censoring):

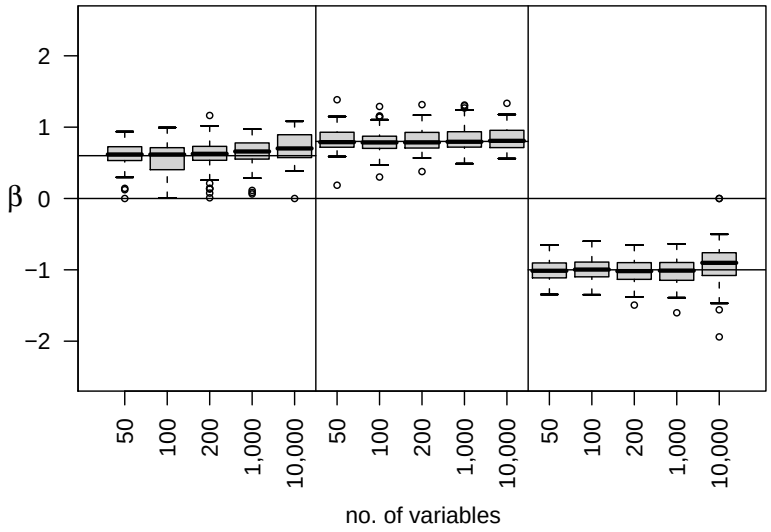
- 100 data sets consisting of $n = 100$ observations from the exponential hazards model $h(t|x) = exp(x^T\beta)$, where the d-dimensional parameter vector β is defined as $\beta = (\beta_1^T, \beta_2^T)^T$,
  - $\beta_1^T = (0.8, -1.0, 0.6)$, $\beta_2^T = 0_{d-3}$
  - $\beta_1^T = (-1.2, -1.0, -0.8, -0.6, -0.4, 0.4, 0.6, 0.8, 1.0)$, $\beta_2^T = 0_{d-10}$

  for $d = 50, 100, 200, 1000, 10000$.
- x_i marginally standard normal with $cor(x_i, x_j) = 0$, $i \neq j$.
- The censoring times were exponentially distributed with mean $U \cdot exp(x^T\beta)$, where U is randomly generated from the uniform distribution over $[1, 3]$ for each simulated data set.

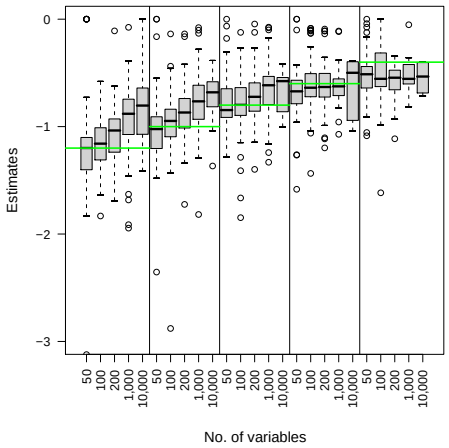
Simulation study: True and false positives (%)

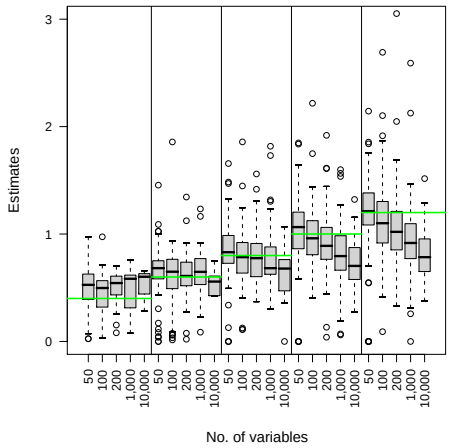


Simulation study: Distribution of estimates



Simulation study: Distribution of estimates





Applications

Real World Situation:

- We observe random variables $(\tilde{T}, \Delta, X)$ for time to event $\tilde{T} = min(T, C)$ and censoring indicator $\Delta = I(T \leq C)$, from some distribution $F_{(\tilde{T},\Delta,X)}$.
- We assume that the conditional censoring distribution $P(C \leq c|Z)$ only depends on the covariates, that is $P(C \leq c|Z) = P(C \leq c|X)$, or, equivalently, that survival time T and censoring time C are conditionally independent given the covariates X.

Assessment of model performance

Let $S(t) = P(\tilde{T} > t)$ denote the marginal event-free probability and $\hat{\pi}(t|x)$ the estimate of conditional survival probabilities $S(t|x)$

Let $Y = I(\tilde{T} > t^*)$ for a fixed time point t^*.

Brier score to measure inaccuracy (Graf et al., 1999)

- Brier score loss function: $\psi(Y, \hat{\pi}) = (Y - \hat{\pi}(t^*|x))^2$
- Brier score for time point t^*: $BS(t^*) = \frac{1}{n}\sum_{i=1}^{n} \psi(y_i, \hat{\pi}(t^*|x_i))$
- **Integrated Brier score**: $IBS(\tau) = \int_0^\tau BS(t)dW(t)$ with weight function $W(t) = 1/\tau$ or $W(t) = (1 - \hat{S}(t))/(1 - \hat{S}(\tau))$.

Algorithms and Software

LASSO coxpath, R package glmpath, version 0.92, 2006/06/06

SCAD R package scad, version 0.53, 2006/05/15 (not released yet).

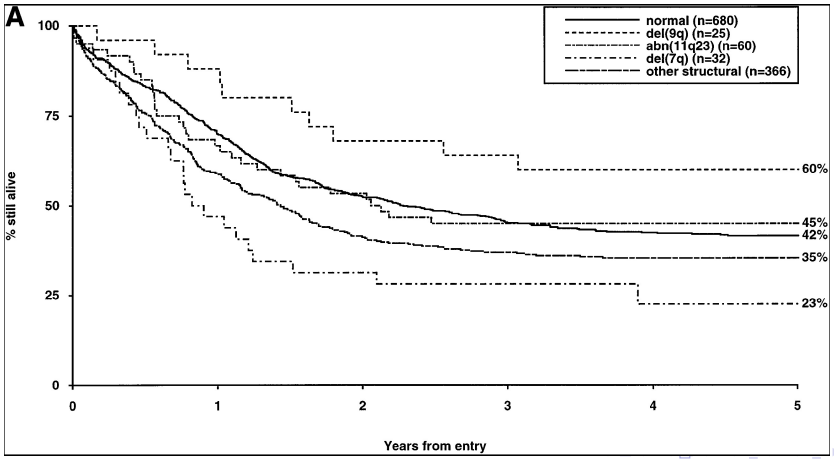
BOOSTING R package mboost, version 0.3-6, 2006/05/10 (not released yet).

Application: AMLSG study

Cytogenetic findings provide a predictive factor in Adult Acute Myeloid Leukemia treatment

The karyotype is used to classify patients as being at

low risk	t(8;21), t(15;17), or inv(16),
intermediate risk	normal karyotype or t(9;11),
high risk	inv(3), -5/del(5q), -7, or complex karyotype [$\geq$ 3 aberrations]



Application: AMLSG study

L. Bullinger et al. (NEJM, 2004)

Use of Gene-Expression Profiling to Identify Prognostic Subclasses in Adult Acute Myeloid Leukemia

- 136 patients with normal karyotype from AML HD98-A (16-60 years) study
  54 peripheral-blood samples and 82 bone marrow specimens
- 42 patients with normal karyotype from AML HD98-B (>60 years) study
  27 peripheral-blood samples and 15 bone marrow specimens
- cDNA microarrays manufactured by the Stanford Functional Genomics Facility

Application: AMLSG study

- 136 patients from AML HD98-A with normal karyotype
- Estimated median follow up was 45 months since first diagnosis.
- Prognostic models were built using clinical data and microarray measurements.

10-fold cross-validation: Integrated Brier score

Method	IBS (3 years follow-up)	Explained variation
Kaplan-Meier	0.1997	-
coxpath		
scad		
glmboost		

Comments

- SVD works for Cox' proportional hazards regression with ridge/scad penalty
- Low bias for SCAD estimates
- Results were comparable with respect to prediction error
- Statistical software for survival analysis in the $d > n$ situation is still "work in progress"

References

Antoniadis, A. Wavelets in Statistics: A Review (with discussion), Journal of the Italian Statistical Association 6 (1997), 97-144.
Breiman, L. Better subset selection using the non-negative garotte. Technometrics 37(1995), 373-384.
Breiman, L. Bagging predictors. Machine Learning 24 (1996), 123-140.
Breiman, L. Random forests. Machine Learning 45 (2001), 5-32.
Bullinger, L., Döhner, K., Bair, E., Fröhling, S., Schlenk, R. F., Tibshirani, R., Döhner, H., and Pollack, J. R. Use of gene-expression profiling to identify prognostic subclasses in adult acute myeloid leukemia. The New England Journal of Medicine 350 (2004), 1605-1616.
Craven, P., and Wahba, G. Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. Numerische Mathematik 31 (1979), 377-403.
Fan, J. Comment on "Wavelets in Statistics: A Review" by A. Antoniadis. Journal of the Italian Statistical Association 6 (1997), 131-138.
Fan, J., and Li, R. Variable selection via nonconcave penalized likelihood and its oracle properties. JASA 96 (2001), 1348-1360.
Fan, J., and Li, R. Variable selection for Cox's proportional hazards model and frailty model. The Annals of Statistics 30 (2002), 74-99.
Frank, I.E., and Friedman, J.H. A statistical view of some chemometrics regression tools. Technometrics 35 (1993), 109-148.
Graf, E., Schmoor, C., Sauerbrei, W., and Schumacher, M. Assessment and comparison of prognostic classification schemes for survival data. Statistics in Medicine 18, 17-18 (1999), 2529-2545.
Gui, J., Li, H. Penalized Cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data. Bioinformatics , 21(2005), 3001-3008.
Hastie, T., and Tibshirani, R. Efficient quadratic regularization for expression arrays. Biostatistics 5 (2004), 329-340.
Hothorn, T., Bühlmann, P., Dudoit, S., Molinaro , A. and van der Laan, M. J. Survival ensembles. Biostatistics (2006) accepted.
Meinshausen, N. Lasso with relaxation. Research report No. 129, ETH Zürich, 2005.
Verweij, P., and van Houwelingen, H. Penalized likelihood in cox regression. Statistics in Medicine 13 (1994), 2427-2436.
Zou, H., and Hastie, T. Regularization and variable selection via the elastic net. J. R. Statist. Soc. B 67 (2005), 301-320.

Attachment: Brier Score for censored data at time point t^*

Three categories contribute to score:
Category 1: $\tilde{T}_i \leq t^*$ and $\Delta_i = 1 \implies (0 - \hat{\pi}(t^*|x))^2$
Category 2: $\tilde{T}_i > t^*$ ($\Delta_i = 1$ or $\Delta_i = 0$) $\implies (1 - \hat{\pi}(t^*|x))^2$
Category 3: $\tilde{T}_i \leq t^*$ and $\Delta_i = 0 \implies$ event status at t^* unknown

Compensate for loss of information by reweighting:
Category 1: weight $1/\hat{G}_T$
Category 2: weight $1/\hat{G}_{t^*}$
Category 3: weight zero

G is Kaplan-Meier estimate of censoring distribution.
Brier score loss function for censored data:

$$\begin{aligned}\psi(y,f) &= (Y - f(x))^2 \\ &= (0 - f(x))^2 I(\tilde{T} \leq t^*, \Delta = 1)(1/\hat{G}_T) \\ &\quad + (1 - f(x))^2 I(\tilde{T} > t^*)(1/\hat{G}_{t^*})\end{aligned}$$

Attachment: Ensemble Learning

Inverse Probability of Censoring Weights

- Here we observe random variables $(\tilde{Y}, \Delta, X)$ where $\tilde{Y} = log(\tilde{T})$ for time to event $\tilde{T} = min(T, C)$ and censoring indicator $\Delta = I(T \leq C)$, from some distribution $F_{(\tilde{Y},\Delta,X)}$.
- Replace the full data loss function $L(Y, \psi(X))$ by an observed data loss function $L(\tilde{Y}, \psi(X)|\eta)$ with nuisance parameter η.
- Inverse probability of censoring weights (IPC weights): the nuisance parameter η is given by the conditional censoring survivor function G

$$L(\tilde{Y}, \psi(X)|G) = L(\tilde{Y}, \psi(X)) \frac{\Delta}{G(\tilde{T}|X)}$$

- Let $\mathbf{w} = (w_1, ..., w_n)$, where $w_i = \Delta_i \hat{G}(\tilde{T}_i|X_i)^{-1}$, denote the IPC weights.

Random Forest for censored data

Step 1 (Initialization). Set $m = 1$ and fix $M > 1$.

Step 2 (Bootstrap). Draw a random vector of case counts $v_m = (v_{m1}, ..., v_{mn})$ from the multinomial distribution with parameters n and $(\sum_{i=1}^{n} w_i)^{-1}\mathbf{w}$.

Step 3 (Base Learner). Construct a partition $\pi_m = (R_{m1}, ..., R_{mK(m)})$ of the sample space X into $K(m)$ cells via a regression tree. The tree is built using the learning sample L with case counts v_m, i.e., is based on a perturbation of the learning sample L with observation i occurring v_{mi} times.

Step 4 (Iteration). Increase m by one and repeat steps 2 and 3 until $m = M$.

- For quadratic loss $L(Y, (X)) = (Y - \psi(X))^2$, the prediction is simply the weighted average of the observed (log)-survival times
- By definition, the weights w_i, and thus the case counts v_{mi} as well as the prediction weights, are zero for censored observations.
- The prediction weights approach is essentially an extension of the classical (unweighted) averaging of predictions extracted from each single partition (cf. Breiman 1996).
- In step 3 of the algorithm the partitions are usually induced by some form of recursive partitioning with additional randomization. This can be implemented by using only a small number of randomly selected covariates for further splitting of every node of the tree.

- Weighted least squares problem

$$\hat{\vartheta}_{\tilde{U},X} = argmin_{\vartheta} \sum_{i=1}^{n} w_i(\tilde{U}_i - h(X_i|\vartheta))^2$$

with pseudo responses

$$U_i = -\frac{\partial L(\tilde{Y}_i, \psi)}{\partial \psi}$$

at $\psi = \hat{f}_m(X_i)$

Generic gradient boosting for censored data

Step 1 (Initialization). Define $\tilde{U}_i = \tilde{Y}_i$ $(i = 1, ..., n)$, set $m = 0$, and $\hat{f}_0(\cdot) = h(\cdot|\hat{\vartheta}_{\tilde{U},X})$. Fix $M > 1$.

Step 2 (Gradient). Compute the residuals

$$\tilde{U}_i = -\frac{\partial L(\tilde{Y}_i, \psi)}{\partial \psi}$$

at $\psi = \hat{f}_m(X_i)$ and fit the base learner $h(\cdot|\hat{\vartheta}_{\tilde{U},X})$ to the new response $\tilde{U}_i$ by weighted least squares.

Step 3 (Update). Update $\hat{f}_{m+1}(\cdot) = \hat{f}_m(\cdot) + \nu h(\cdot|\hat{\vartheta}_{\tilde{U},X})$ with step size $0 < \nu \leq 1$.

Step 4 (Iteration). Increase m by one and repeat steps 2 and 3 until $m = M$.

Note, that the number of iterations, M, is a tuning parameter, which needs to be determined via cross-validation.

Attachment: Oracle Property

$\hat{\beta} = (\hat{\beta}_1^T, \hat{\beta}_2^T)^T$ satisfyies

(a) Sparsity: $\hat{\beta}_2 = 0$

(b) Asymptotic normality:

$$\sqrt{n}(I_1(\beta_{10}) + \Sigma)\left\{\hat{\beta}_1 - \beta_{10} + (I_1(\beta_{10}) + \Sigma)^{-1} b\right\} \rightarrow \mathcal{N}(0, I_1(\beta_{10}))$$

in distribution where $I_1(\beta_{10}) = I_1(\beta_{10}, 0)$, the Fisher information knowing $\beta_2 = 0$.
Here $b = (p'_\lambda(|\beta_{10}|)sgn(\beta_{10}), ..., p'_\lambda(|\beta_{s0}|)sgn(\beta_{s0}))^T$ and s is the number of components of β_{10}.

For more details see Fan & Li (2001).