LOSS BASED CROSS-VALIDATED DELETION/SUBSTITUTION/ADDITION ALGORITHMS IN LEARNING: Applications in Genomics

Mark van der Laan

www.stat.berkeley.edu/ laan

Joint work with Sandra Sinisi, Annette Molinaro, Sandrine Dudoit, Merrill Birkner, Sunduz Keles.

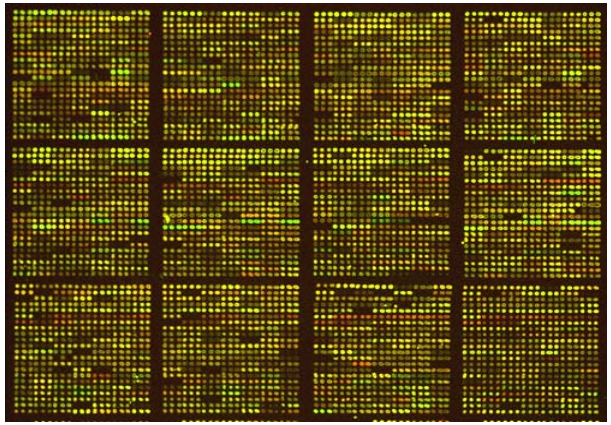
Division of Biostatistics, University of California, Berkeley.

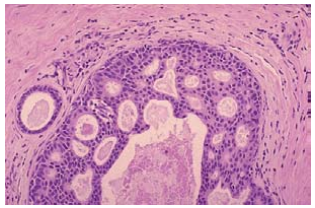
www.bepress.com/ucbbiostat/

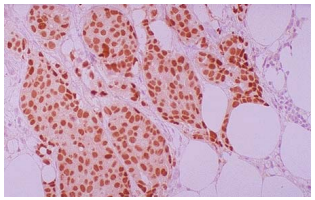
Taipei Symposium on Statistical Genomics, Dec 14-18, 2004

HIGH DIMENSIONAL DATA STRUCTURES

Covariates



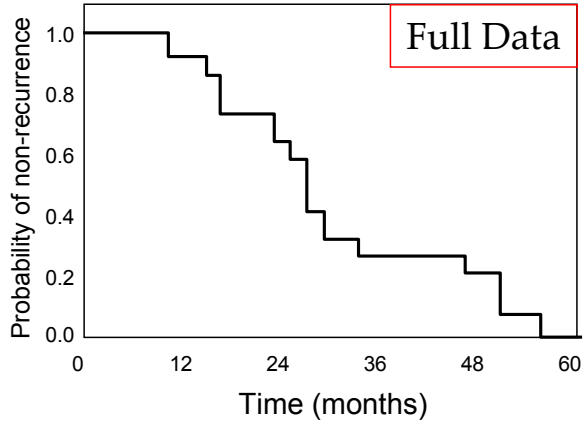


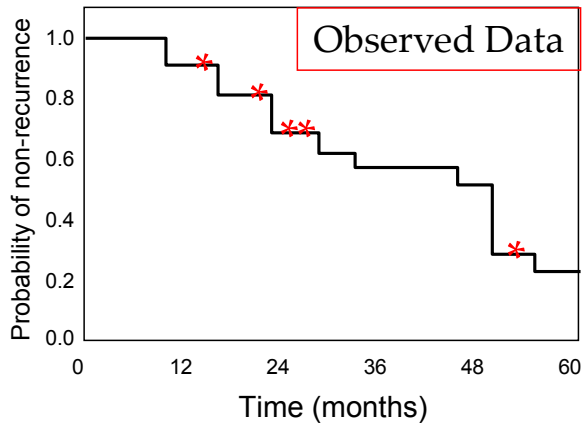


QUESTIONNAIRE

Age___
Age at First Child_____ Breast Feeding_____
Diet_____
Hormone Replacement Therapy_____
Family History_____
Smoking History_____
Education_____
Income_____ Birth Control_____

Clinical Outcomes





ESTIMATION PROBLEMS

1. Prediction of clinical outcomes based on epidemiological and genomic data such as gene expression/ single nucleotide polymorphism (SNP)/ comparative genomic hybridization (CGH).
2. *Prediction of HIV-1 replication capacity from sequence data*

   All major anti-retrovirals used in HIV-1 treatment have targeted either reverse transciptase or protease. Mutations in the HIV-virus which confer resistance to these compounds are important for clinical decision making. Prediction of gene-expression from regulatory-DNA-sequence.
3. Estimation of causal effects of exposure/treatment regimes on outcomes, such as asthma-development, CD4-count, survival, in longitudinal studies with high dimensional covariates.

Dominating features: High dimensional data and parameter space, censored clinical outcomes such as survival, and time till

recurrence.

CONCERNS REGARDING CURRENT ALGORITHMS

1) We need algorithms which data adaptively select variables and functional forms of these variables (e.g. 2-way interaction, 3 way-interaction etc.), i.e. working models, **without the need to a-priori having to enumerate all possible functional forms**.

2) Current algorithms (with the important exception of **Logic Regression**) in prediction (e.g., CART, MARS, STEP-AIC) use forward/backward variable selection, and typically only use cross-validation to select size. Consequently, these algorithms yield **limited searches** among candidate estimators.

3) Lack of general framework and theory for handling censored data and general estimation problems.

UNIFIED LOSS BASED ESTIMATION: ROAD MAP

DATA and MODEL We have n i.i.d observations $O_1, \ldots, O_n$ of $O \sim P_0$ and a model $\mathcal{M}$ for P_0. Let P_n be the empirical distribution.

PARAMETER and LOSS FUNCTION Define the parameter of interest of P_0, $\psi_0 : \mathcal{S} \to \mathbb{R}$, as the minimizer over the parameter space $\mathbf{\Psi}$ of a so called risk function defined as the expectation of a loss function $(O, \psi) \to L(O, \psi)$ of the observed experimental unit O and a candidate parameter value ψ: $\psi_0 = \arg\min_{\psi \in \mathbf{\Psi}} E_0 L(O, \psi)$. The loss function is allowed to depend on true data generating distribution (i.e., unknown).

Parameter	Data	Loss
$\psi_0(W) = E_0(Y \mid W)$	$O = (Y, W)$	$L(O, \psi) = (Y - \psi(W))^2$
$\psi_0(W) = MED_0(Y \mid W)$	$O = (Y, W)$	$L(O, \psi) = \mid Y - \psi(W) \mid$
$\psi_0(Y \mid W) = f_0(Y \mid W)$	$O = (Y, W)$	$L(O, \psi) = -\log \psi(Y \mid W)$

CENSORED DATA LOSS FUNCTIONS

Suppose $O = \Phi(C, X) \sim P_{Q_{X0}, G_0}$ is censored data structure, where $G_0(\cdot \mid X)$ (satisfying coarsening at random assumption) and Q_{X0}, denote the censoring mechanism and identified part of the full data distribution, respectively. Suppose $\psi_0 = \Psi(Q_{X0})$ is a parameter of full data distribution with corresponding full data loss function $L(X, \psi)$.

Use general estimating function methodology in van der Laan & Robins (2002) to map (with the IPCW or the optimal DR-IPCW mapping) the full data loss function, $L(X, \psi)$, into an observed data loss function, $L(O, \psi \mid G, Q)$, with the same risk

$$\underbrace{\int L(o, \psi \mid G_0, Q_0) dP_0(o)}_{\text{Observed Data Risk}} = \underbrace{\int L(x, \psi) dF_{X,0}(x)}_{\text{Full Data Risk}}.$$

PREDICTION OF SURVIVAL

Let T be a log-survival time, and suppose that our goal is to estimate the optimal predictor $\psi_0(W) = E_0(T \mid W)$. However, due to right-censoring by a variable C, we only observe $O_i = (\tilde{T}_i \equiv \min(T_i, C_i), \Delta_i = I(T_i \leq C_i), W_i)$.

Censoring mechanism: Let $G(\cdot \mid T, W)$ be the conditional distribution of censoring C, given (T, W).

CAR: Assume that censoring is independent of survival time, given W: i.e., $G(\cdot \mid T, W) = G(\cdot \mid W)$.

The **Inverse Probability of Censoring Weighted** (IPCW) Squared Error Loss Function

$$L(O, \psi \mid G) \equiv L(T, W, \psi) \frac{\Delta}{P_G(\Delta = 1 \mid W)} = (T - \psi(W))^2 \frac{\Delta}{\bar{G}(T \mid W)}.$$

For the optimal (that is, minimal variance, and maximally robust) **double robust IPCW loss function**, we refer to Robins, Rotnitzky (1992), van der Laan, Robins (2002).

Data	Loss
$X = (T, W)$	$L(X, \psi) = (T - \psi(W))^2$
$O = (\tilde{T} = \min(T, C), \Delta, W)$	$L(O, \psi \mid G_0) = (T - \psi(W))^2 \frac{\Delta}{\bar{G}_0(T \mid W)}.$

GENERAL CONSTRUCTION OF LOSS FUNCTION

Identify wished risk function: $\psi \to \Theta(\psi \mid P_0)$ is minimized at ψ_0.

Determine corresponding optimal loss function: Set loss function at ψ equal to the (risk-function centered) efficient influence curve of the real valued parameter $\Theta(\psi \mid P_0)$ of P_0 in model $\mathcal{M}$.

ROAD MAP: PARAMETRIZE

We describe any of the allowed functions in $\mathbf{\Psi}$ with linear combinations $\psi_{I,\beta}(w) = \sum_{j \in I} \beta_j \Phi_j(w)$ of basis functions $w \to \Phi_j(w)$ indexed by an index set I. For simplicity, we consider the case that the basis functions do not depend on unknown parameters.

Tensor products of univariate basis functions For example, if we use a polynomial basis, then for each integer vector $\vec{p} = (p_1, \ldots, p_d)$, we have a basis function $\phi_{\vec{p}}(W) = W_1^{p_1} \cdots W_d^{p_d}$. Each index set $I = \{\vec{p}_1, \ldots, \vec{p}_k\}$, corresponds now with a linear subspace in variables being tensor products of polynomial powers.

Indicators of sets of a partition: Given a region R in $\mathcal{S}$, let $\Phi_R(\cdot) = I(\cdot \in R)$ be the indicator of this region. Each partition $I = \{R_1, \ldots, R_k\}$ of $\mathcal{S}$ corresponds with a linear subspace in variables being indicators of sets R_j: thus, a histogram subspace.

ROAD MAP: SELECTING A SIEVE

One obtains a collection of subspaces $\mathbf{\Psi}_s \subset \mathbf{\Psi}$ by constraining the subset of basis functions. Let $\mathbf{\Psi}_{s,\delta} \subset \mathbf{\Psi}_s$ be obtained by also restricting the vector of coefficients. For example, for each choice of $(s = (k, k_1), \delta)$, $\mathbf{\Psi}_{s,\delta}$ denotes the functions $\psi_{I,\beta} = \sum_{j \in I} \beta_j \Phi_j$ for which

1) the size of I is smaller or equal than k,

2) each basis function Φ_j, $j \in I$, has a complexity measure (e.g. number of terms in tensor product) smaller or equal than k_1,

3) the norm $\| \beta \|$ of β (e.g. euclidean, manhattan) is smaller or equal than δ, or β is constrained to be on a δ-grid).

ROAD MAP: MINIMUM EMPIRICAL RISK ESTIMATOR

For each subspace compute minimum empirical risk estimator, $\psi_{s,\delta,n} = \arg\min_{\psi_{I,\beta} \in \mathbf{\Psi}_{s,\delta}} \sum_i L(O_i, \psi_{I,\beta})$, which is carried out by

1. Given a subset I of basis functions, minimize empirical risk over the corresponding coefficients β, which gives us a subset-specific estimator, $\hat{\Psi}_I(P_n)$.

2. Minimize empirical risk $f_E(I) = \sum_i L(O_i, \hat{\Psi}_I(P_n))$ of the subset-specific estimator over the allowed subsets of basis functions I: Deletion/Substitution/Addition algorithm. For example, in regression with the squared error loss function:

$$f_E(I) \equiv \sum_{i=1}^{n} (Y_i - \hat{\Psi}_I(P_n)(W_i))^2.$$

ROAD MAP: THE CROSS-VALIDATION SELECTOR

Let P_n be the empirical distribution of the observed sample $O_1, \ldots, O_n$.

Estimator: An estimator of ψ_0 is a mapping (i.e., an algorithm) from P_n into a particular element of $\mathbf{\Psi}$. Notation: $P_n \to \hat{\Psi}(P_n) \in \mathbf{\Psi}$.

Empirical risk estimate: Given an estimator $\hat{\Psi}(P_n)$, the empirical conditional risk estimate is simply the empirical mean of the loss $L(O, \hat{\Psi}(P_n)) = (Y - \hat{\Psi}(P_n)(W))^2$:

$$\frac{1}{n} \sum_{i=1}^{n} L(O_i, \hat{\Psi}(P_n)).$$

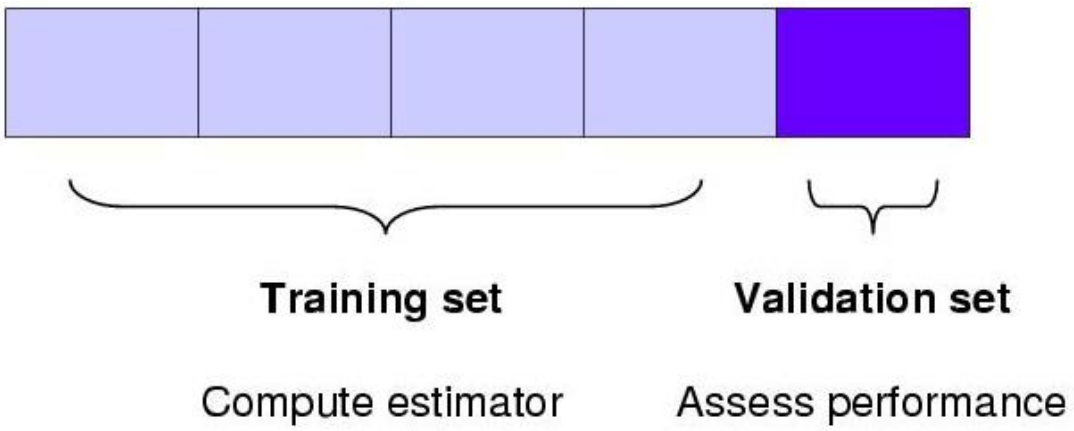
Cross-validated conditional risk estimate: In this case, one applies the estimator to a part of the sample (training sample) and one computes the average loss of the obtained estimator over the remaining sample (validation sample). One averages this risk

estimate over a particular number of splits of the sample.

Formally, define a random vector $S_n \in \{0,1\}^n$ for splitting the sample into a validation and a training sample.

$$S_{n,i} = \begin{cases} 0 & \text{if} \quad \text{i-th observation is in the training sample} \\ 1 & \text{if} \quad \text{i-th observation is in the validation sample} \end{cases}$$

Different choices of S_n cover all types of cross-validation schemes including $V-$ fold cross-validation, monte carlo cross validation, and bootstrap cross-validation. For example, in 5-fold cross-validation S_n has 5 possible outcomes.



Cross-validation selector: The cross-validation selector (s_n, δ_n) of (s, δ) chooses the estimator minimizing the cross-validated risk estimate.

RISK DISSIMILARITY

We define the risk dissimilarity as

$$d(\psi, \psi_0) \equiv \int \{L(O, \psi) - L(O, \psi_0)\} \, dP_0(O).$$

Data	Loss	$d(\psi, \psi_0)$
$O = (Y, W)$	$L(O, \psi) = (Y - \psi(W))^2$	$\int (\psi - \psi_0)^2(W) dP_0(W)$
$O = (Y, W)$	$L(O, \psi) = -\log \psi(Y \mid W)$	$-\int \log \frac{\psi}{\psi_0}(Y \mid W) dP_0(Y, W)$

CROSS-VALIDATION VERSUS ORACLE SELECTOR

Given a collection of K_n estimators $P_n \to \hat{\Psi}_k(P_n)$, we have proved the following inequality for the risk dissimilarity of the estimator selected by cross-validation: for any $\lambda > 0$

$$\begin{aligned} E d(\hat{\Psi}_{\hat{k}(CV)}(P_{n(1-p)}), \psi_0) \quad &\leq \quad (1+2\lambda) E \min_k d(\hat{\Psi}_k(P_{n(1-p)}), \psi_0) \\ &\quad + C(\lambda) \frac{\log K(n)}{np}, \end{aligned}$$

where $C(\lambda)$ is an explicit function of the uniform norm of the loss function, and p is the proportion of the sample used for the validation samples.

For example, if $n = 1000$, and one uses 10-fold cross validation (i.e, $p = 0.1$), and one wishes $\log K(n)/np \leq 0.1$, then one can consider $K(n) = 22026$ estimators.

Consequently, if $K(n) = (np)^m$, then the cross-validation selector is either asymptotically equivalent with the oracle selector, or achieves the almost parametric rate $\log n/n$.

THEORETICAL FOUNDATION OF ROAD MAP

Let $\mathbf{\Psi}_{s,\delta} \subset \mathbf{\Psi}_s$ be obtained by restricting the coefficients in front of the basis functions to be on a δ-grid, and let $N_s(\delta)$ be the number of grid points. Let $\hat{\Psi}_{\delta,s}(P_n)$ be the minimum empirical risk estimator over this discrete set, and let $(\hat{\delta}, \hat{s})$ be the cross-validation selector of (δ, s) defined above, among a set of $K(n)$ candidates. For each s and δ, consider the approximation error of $\mathbf{\Psi}_{s,\delta}$:

$$B_0(s,\delta) \quad = \quad \min_{\psi \in \mathbf{\Psi}_{s,\delta}} \int L(O,\psi) - L(O,\psi_0) dP_0(O)$$

We have for any $\lambda > 0$

$$Ed(\hat{\Psi}_{\hat{\delta},\hat{s}}(P_{n(1-p)}), \psi_0) \leq (1+2\lambda)*$$

$$\min_s \min_\delta \left\{ (1+2\lambda) B_0(s,\delta) + 2C(M_1, M_2, \lambda) \frac{1 + \log(N_s(\delta))}{n(1-p)} \right\}$$

$$+2C(M_1, M_2, \lambda) \frac{1 + \log(K(n))}{np},$$

where $C(M_1, M_2, \lambda) = 2(1+\lambda)^2(M_1/3 + M_2/\lambda)$,

$$M_1 = \sup_{O,\psi} L(O,\psi) - L(O,\psi_0)$$

$$M_2 = \sup_{\psi} \frac{\mathrm{VAR}(L(O,\psi) - L(O,\psi_0))}{\mathrm{E}(L(O,\psi) - L(O,\psi_0))}.$$

This implies, that the estimator achieves the minimax adaptive optimal rate of convergece w.r.t. to the sieve $(\mathbf{\Psi}_s : s)$:

$$Ed(\psi_{\hat{\delta},\hat{s}}(P_n), \psi_0) = \min_{\{s:\psi_0 \in \Psi_s\}} O(R_n(s)) + O\left(\frac{\log K(n)}{n}\right),$$

where $R_n(s)$ is the minimax optimal rate of convergence for $d(\hat{\psi}, \psi_0)$ for the subspace $\mathbf{\Psi}_s$. That is, the estimator $\hat{\psi}_{\hat{\delta},\hat{s}}$ achieves the optimal rate of convergence for the smallest subspace containing the true ψ_0.

LARS/LASSSO VERSUS ϵ-NET ESTIMATOR: SIMULATION

We simulate data sets from a linear regression $Y \sim \beta X + N(0, \sigma^2)$ with $X(j) \sim U(0,1)$, $j = 1, .., 10$, $\sigma^2 \in \{1, 2, 10\}$, and uniformly distributed regression coefficients β. We generated 10 simulated data sets of various sample sizes, and compared the ϵ-net linear regression estimator of β with the Least-Angle-Linear Regression estimator, (lars), and Lasso based on residual sum of squares on an independent sample of 10,000 observations.

	mean.eps.net	sd.eps.net	mean.lasso	sd.lasso
20	2.11	0.33	111.70	182.26
30	1.53	0.22	3.80	2.91
40	1.70	0.44	1.50	0.22
50	1.21	0.06	1.96	0.35
70	1.18	0.04	2.96	1.51
90	1.17	0.10	1.49	0.07
120	1.04	0.01	1.12	0.09

Table 1: dist = uniform, sigma = 1, V = 5

	mean.eps.net	sd.eps.net	mean.lasso	sd.lasso
20	196.25	28.69	603.04	208.31
30	142.14	20.48	222.86	102.17
40	135.03	12.37	166.42	19.48
50	111.96	1.38	215.66	37.49
70	115.69	5.53	346.72	189.70
90	116.22	3.85	149.31	10.22
120	110.33	6.01	109.00	3.87

Table 2: dist = uniform, sigma = 10, V = 5

DELETION/SUBSTITUTION/ADDITION ALGORITHM

The D/S/A algorithm aims to minimize a function $I \rightarrow f_E(I)$ (e.g., f_{RSS}) over subsets of basis functions, and is defined by three set functions $DEL(I_0)$, $SUB(I_0)$, and $ADD(I_0)$, which maps a current subset I_0 into a collection of subsets of size $\mid I_0 \mid -1$ (deletion moves), $\mid I_0 \mid$ (substitution moves), and $\mid I_0 \mid +1$ (addition moves), respectively.

ALGORITHM

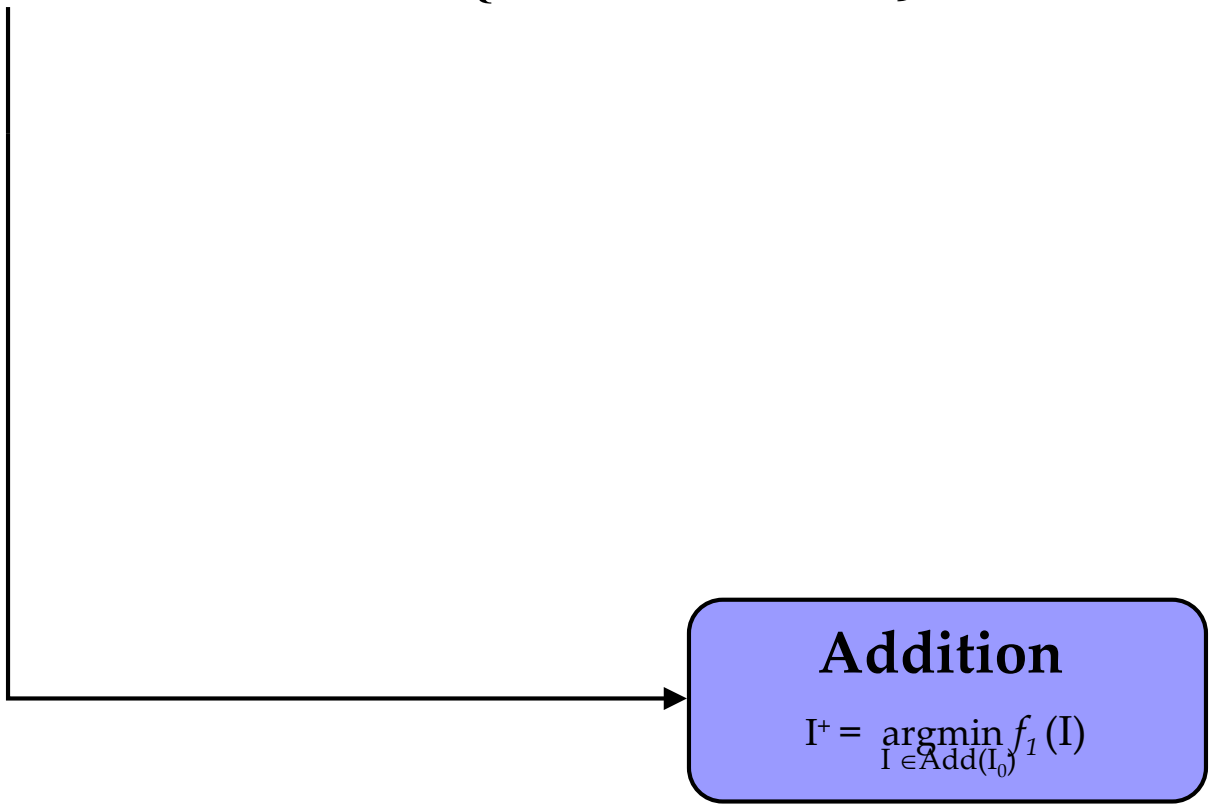
$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$

ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$



ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$

$I_0 = I^+,$

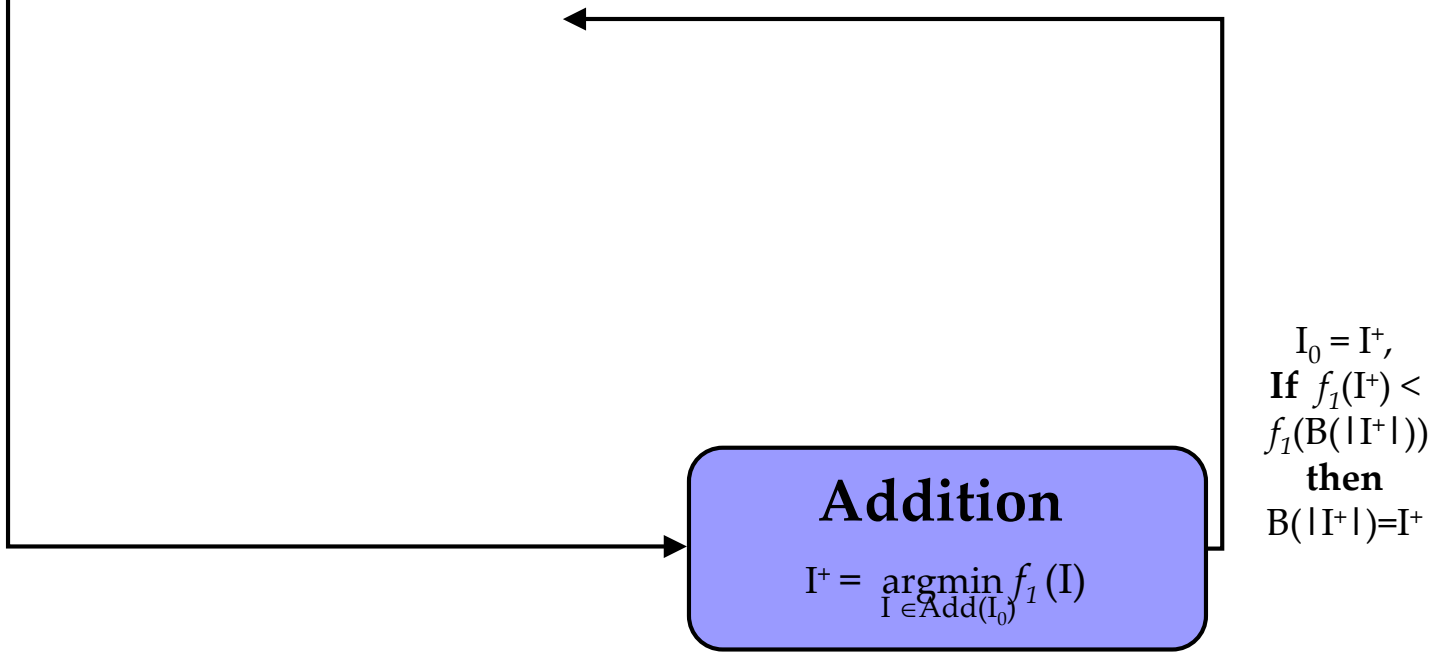
Addition

$I^+ = \underset{I \in Add(I_0)}{\operatorname{argmin}} f_1(I)$

ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

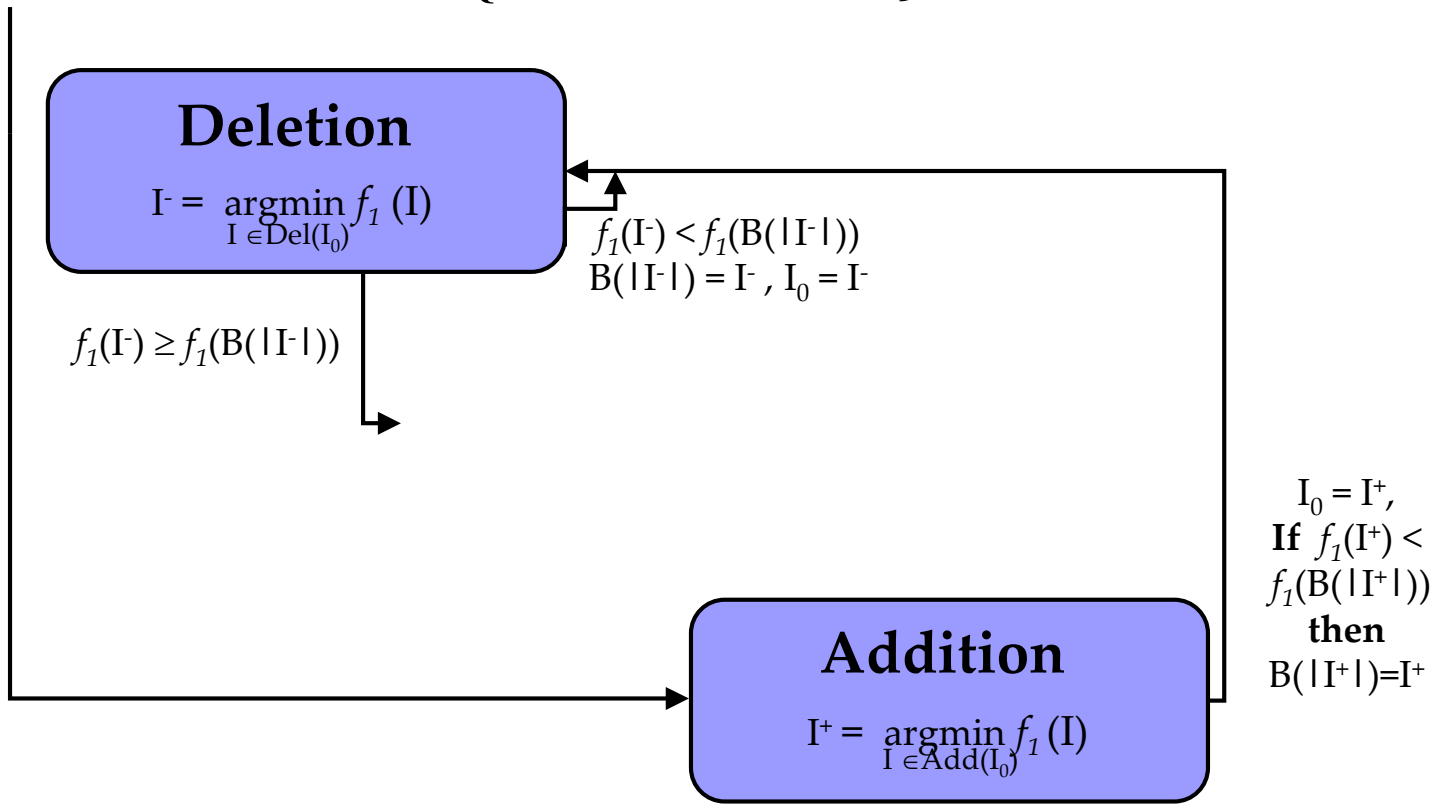
Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$



ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

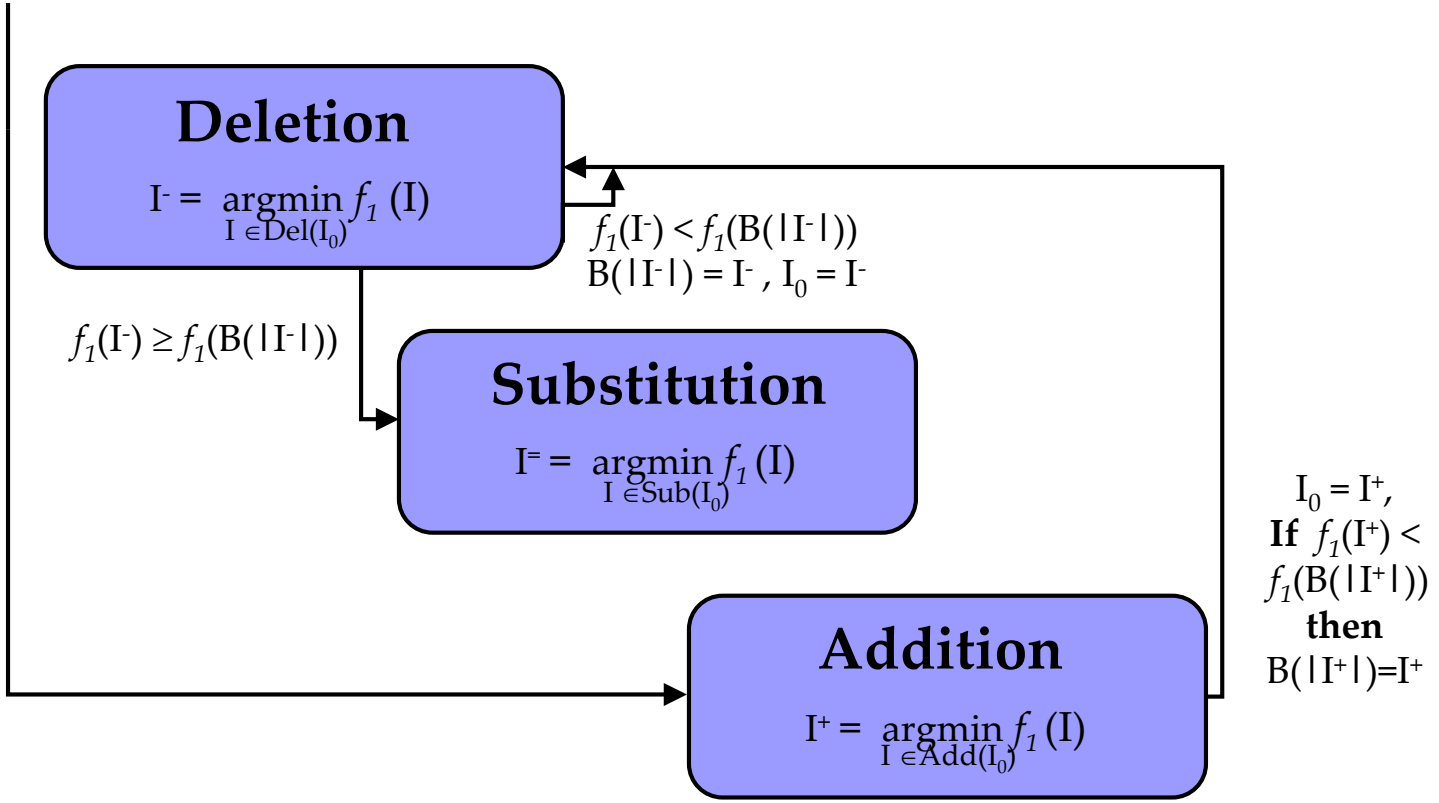
Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$



ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

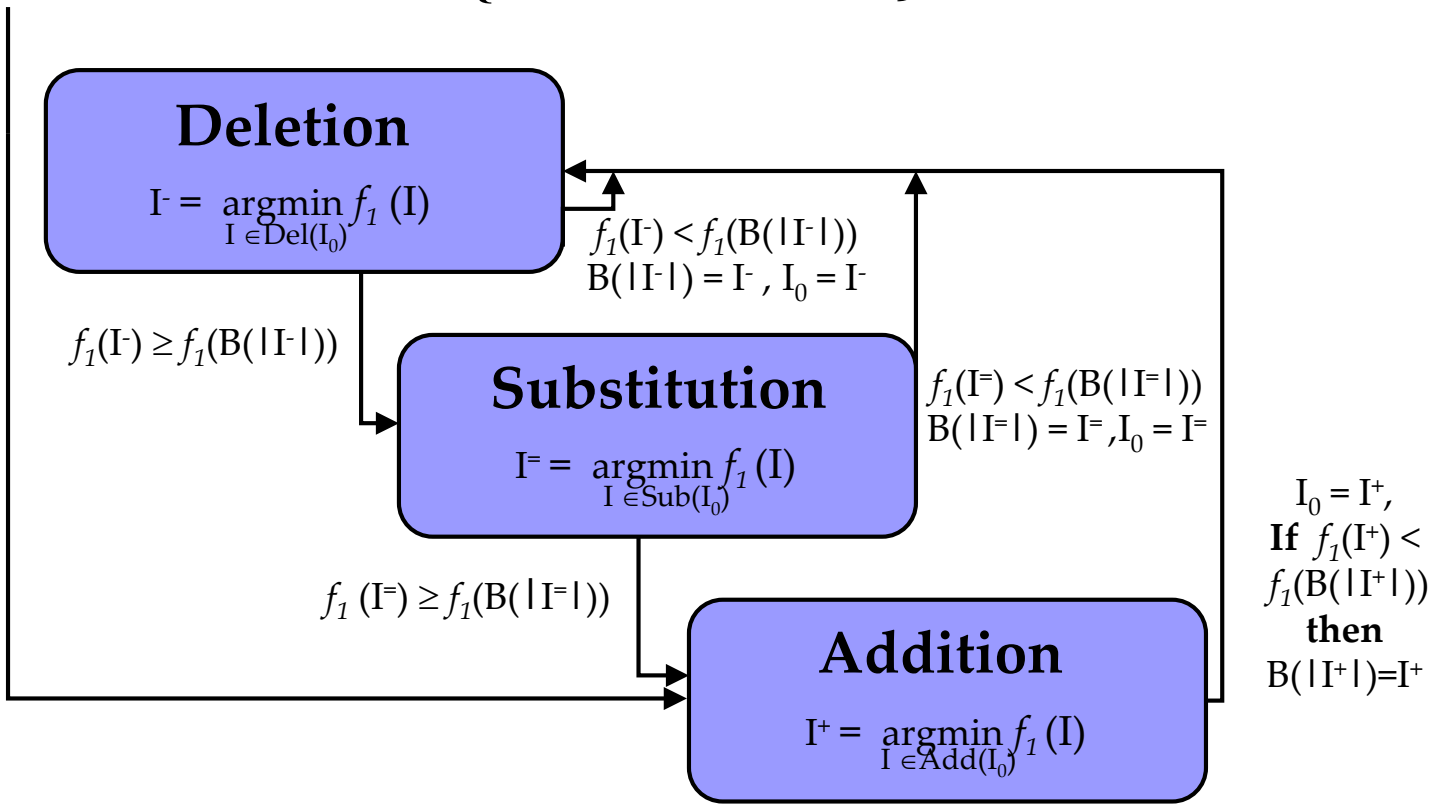
Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$



ALGORITHM

$f_1(I) = \Sigma_{i=1}^{n} L(O_i, \psi_I(\,|P_n))$

Initiate Algorithm $\{\ I_0\ ,\ B(k),\ \text{where } k = 0,\ldots,M\ \}$



PROPOSAL FOR TENSOR PRODUCT MOVES

Deletion moves. $DEL(I_0)$ maps into the k subsets of size $k-1$ corresponding with deleting one of the k basis functions in I_0.

Substitution moves. Given a basis function indexed by $\vec{p} \in I_0$, replace it by the basis function indexed by $\vec{p} \pm \vec{e}_j$, where $\vec{e}_j$ denotes the j-th unit vector, $j = 1, \ldots, d$. If this move causes $\vec{p}$ to have more than k_1 non-zeros (being a particular constrained complexity measure of the basis function), then we transform it into a set of k_1 **swap moves**, by setting one of the non-zero components equal to zero.

Illustration, excluding swap moves:

$$\vec{p} \rightarrow \begin{cases} (p_1+1, p_2, p_3, \ldots, p_d) \\ (p_1, p_2+1, p_3, \ldots, p_d) \\ \vdots \\ (p_1, p_2, p_3, \ldots, p_d+1) \\ (p_1-1, p_2, p_3, \ldots, p_d) \\ (p_1, p_2-1, p_3, \ldots, p_d) \\ \vdots \\ (p_1, p_2, p_3, \ldots, p_d-1) \end{cases}$$

for each $\vec{p} \in I_0$.

Addition moves. Given the current index set I_0, the addition moves are obtained by adding to I_0 the basis functions indexed by one of the unit vectors or by one of the basis functions in $SUB(I_0)$.

Illustration:

$$\vec{p}_{k+1} = \begin{cases} (1,0,\ldots,0) \\ \vdots \\ (0,\ldots,0,1) \\ (p_1+1,p_2,p_3,\ldots,p_d) \\ \vdots \\ (p_1,p_2,p_3,\ldots,p_d+1) \\ (p_1-1,p_2,p_3,\ldots,p_d) \\ \vdots \\ (p_1,p_2,p_3,\ldots,p_d-1) \end{cases}$$

SIMPLE EXAMPLE FOR POLYNOMIAL BASIS

Regression: Let $d = 4$ and $Y = W_1 W_2 W_3 + W_2 W_4^5 + \varepsilon$. Then $k = 2$, $\vec{p}_1 = (1, 1, 1, 0)$, $\vec{p}_2 = (0, 1, 0, 5)$.

A deletion move simply means removing one of the terms of the current model and fitting a model of size $k - 1$.

The substitution moves involve replacing the s^{th} term for $s = 1, \ldots, k$ with a new term, keeping the size of the model fixed at k.

The possible substitution moves are given by:

$$
SUB(I_0) = \left\{ \begin{array}{ll}
W_1^2 W_2 W_3 + W_2 W_4^5 & \vec{p}_1 = (2,1,1,0) \\
W_1 W_2^2 W_3 + W_2 W_4^5 & \vec{p}_1 = (1,2,1,0) \\
W_1 W_2 W_3^2 + W_2 W_4^5 & \vec{p}_1 = (1,1,2,0) \\
W_1 W_2 W_3 W_4 + W_2 W_4^5 & \vec{p}_1 = (1,1,1,1) \\
W_2 W_3 + W_2 W_4^5 & \vec{p}_1 = (0,1,1,0) \\
W_1 W_3 + W_2 W_4^5 & \vec{p}_1 = (1,0,1,0) \\
W_1 W_2 + W_2 W_4^5 & \vec{p}_1 = (1,1,0,0) \\
 & \\
W_1 W_2 W_4^5 + W_1 W_2 W_3 & \vec{p}_2 = (1,1,0,5) \\
W_2^2 W_4^5 + W_1 W_2 W_3 & \vec{p}_2 = (0,2,0,5) \\
W_2 W_3 W_4^5 + W_1 W_2 W_3 & \vec{p}_2 = (0,1,1,5) \\
W_2 W_4^6 + W_1 W_2 W_3 & \vec{p}_2 = (0,1,0,6) \\
W_4^5 + W_1 W_2 W_3 & \vec{p}_2 = (0,0,0,5) \\
W_2 W_4^4 + W_1 W_2 W_3 & \vec{p}_2 = (0,1,0,4)
\end{array} \right.
$$

We also note that, if the total number of terms in the tensor products is bounded by (e.g.) $k_2 = 3$, then the unallowed substitution move $W_1W_2W_3W_4 + W_2W_4^5$, would be replaced by these swap moves:

- $W_2W_3W_4 + W_2W_4^5$,
- $W_1W_3W_4 + W_2W_4^5$,
- $W_1W_2W_4 + W_2W_4^5$.

The additions set, $ADD(I_0)$, contains $13 + 4 = 17$ index sets, $I^+ = \{\vec{p}_1, \vec{p}_2, \vec{p}_3\}$, of size $k = 3$, where $\vec{p}_3$ is either one of the thirteen vectors introduced in the above substitutions set $SUB(I_0)$ or one of the four unit vectors $\vec{u}_j$, $j = 1, \ldots, 4$. Specifically, the 17 possible additions are:

$$
ADD(I_0) = \begin{cases}
W_1 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,0,0,0) \\
W_2 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,1,0,0) \\
W_3 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,0,1,0) \\
W_4 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,0,0,1) \\
\\
W_1^2W_2W_3 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (2,1,1,0) \\
W_1W_2^2W_3 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,2,1,0) \\
W_1W_2W_3^2 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,1,2,0) \\
W_1W_2W_3W_4 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,1,1,1) \\
W_2W_3 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,1,1,0) \\
W_1W_3 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,0,1,0) \\
W_1W_2 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,1,0,0) \\
\\
W_1W_2W_4^5 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (1,1,0,5) \\
W_2^2W_4^5 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,2,0,5) \\
W_2W_3W_4^5 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,1,1,5) \\
W_2W_4^6 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,1,0,6) \\
W_4^5 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,0,0,5) \\
W_2W_4^4 + W_1W_2W_3 + W_2W_4^5 & \vec{p}_3 = (0,1,0,4)
\end{cases}
$$

DOES THE DSA ALGORITHM DO THE JOB?

Is the algorithm capable to find the global minimum (i.e., the optimal predictor $W \to \psi_0(W) = E_0(Y \mid W)$) when n is large enough?

We generated $n = 1000$ observations from the following three true regression models with zero error, $d = 100$, $X_j \sim U(0,1)$, and we check if the D/S/A algorithm finds the truth.

$E_1[Y|X] = X_1X_{12}X_{13}^2X_{22}X_{24}X_{54}X_{79}X_{83}X_{95} + X_{15}X_{18}X_{37}X_{42}X_{68} + X_6X_{22}X_{33}^3X_{40}X_{58}X_{75}X_{82}X_{87} + X_{15}X_{31}$

$E_2[Y|X] =$

$X_7X_{25}X_{31}X_{59}X_{63}X_{68}X_{70}X_{83}X_{88}X_{98} + X_0X_{32}X_{47}X_{54}X_{66}X_{72}X_{73}X_{77} + X_{82} + X_7X_{49}X_{55}X_{73}X_{80} + X_{33}X_{40} + X_{18}X_{21}X_{40}X_{56}X_{59}X_{71}X_{91} + X_9X_{13}X_{18}X_{20}X_{41}X_{53}X_{69}X_{95} + X_3X_{38}X_{78}X_{96} + X_0X_{20}X_{64}X_{88}X_{91}X_{96} + X_2X_6X_{16}X_{37}X_{45}X_{46}X_{61}X_{68}X_{91}X_{95}$

$$E_3[Y|X] = X_0X_1^2X_4^4X_{99}^{10} + X_{45} +$$
$$X_2^2X_8X_{14}X_{20}X_{22}X_{29}X_{36}X_{39}X_{41}X_{44}X_{48}X_{56}X_{62}X_{63}X_{65}X_{87} +$$
$$X_{27}X_{48}X_{63}X_{77}X_{78}X_{93}X_{94} + X_{71} +$$
$$X_{12}X_{18}X_{22}X_{44}X_{50}X_{55}X_{57}X_{64}X_{73}^2X_{80}X_{83}X_{93}X_{94}X_{96} + X_{69}X_{91} +$$
$$X_2X_4X_{22}X_{23}X_{28}X_{36}X_{53}X_{79}X_{88} + X_{48}X_{70}X_{82}X_{97} +$$
$$X_3X_{24}X_{29}X_{54}X_{64}X_{80}$$

$$E_4[Y|X] = X_0X_1^2X_2X_3^2 \ldots X_{99}^2X_{100}$$

Simulation Results for Four Models

Zero error

$E[Y\|X]$	X	n	d	$Sens$	$Spec$	RSS
$E_1[Y\|X]$	$\mathcal{U}(0,1)$	1000	100	1.0	1.0	0.000000
$E_2[Y\|X]$	$\mathcal{U}(0,1)$	1000	100	1.0	1.0	0.000000
$E_3[Y\|X]$	$\mathcal{U}(0,1)$	1000	100	1.0	1.0	0.000000
$E_4[Y\|X]$	$\mathcal{U}(0,1)$	1000	100	1.0	1.0	0.000000

Simulation II: Comparison to forward selection

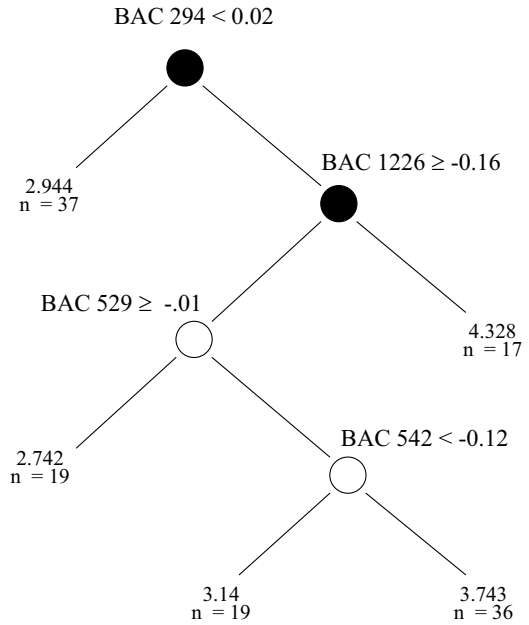
- Run D/S/A algorithm
- Run a forward selection (i.e., A) that uses our addition moves
- Choose size with CV.

Comparing the D/S/A to a forward selection algorithm

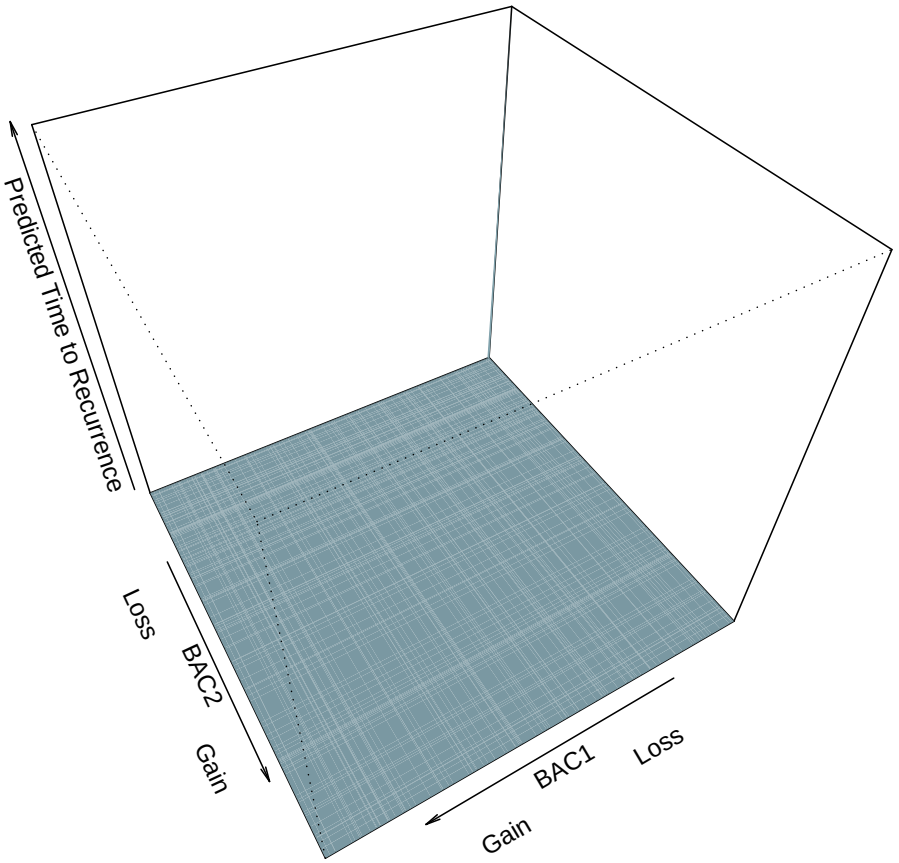
- Data simulated from $y = \sum_{j=1}^{5} c_j w^{2j} + er$, where $w \sim U(-1, 1)$, $er \sim N(0, \sigma = .25)$ and $c_j = 1/j^2$.
- We generated 50 repetitions for three sample sizes (col 1), three v-fold cross-validations (col 2), and each time computed our CV-DSA algorithm and CV-forward selection (col 3).
- The true risks of the estimators (cols 4-9) are based on an independent test sample of $n = 10000$. col 4 is the average of the 50 risks (with the L_2 loss function) for each method, col 5 is the standard deviation of the risks over the 50 reps, col 6 is the average size (number of basis functions), col 7 is the sensitivity and specificity, col 8 is the ratio of averaged risks (col 4) $-$ optimal risk, (ours/fscv).

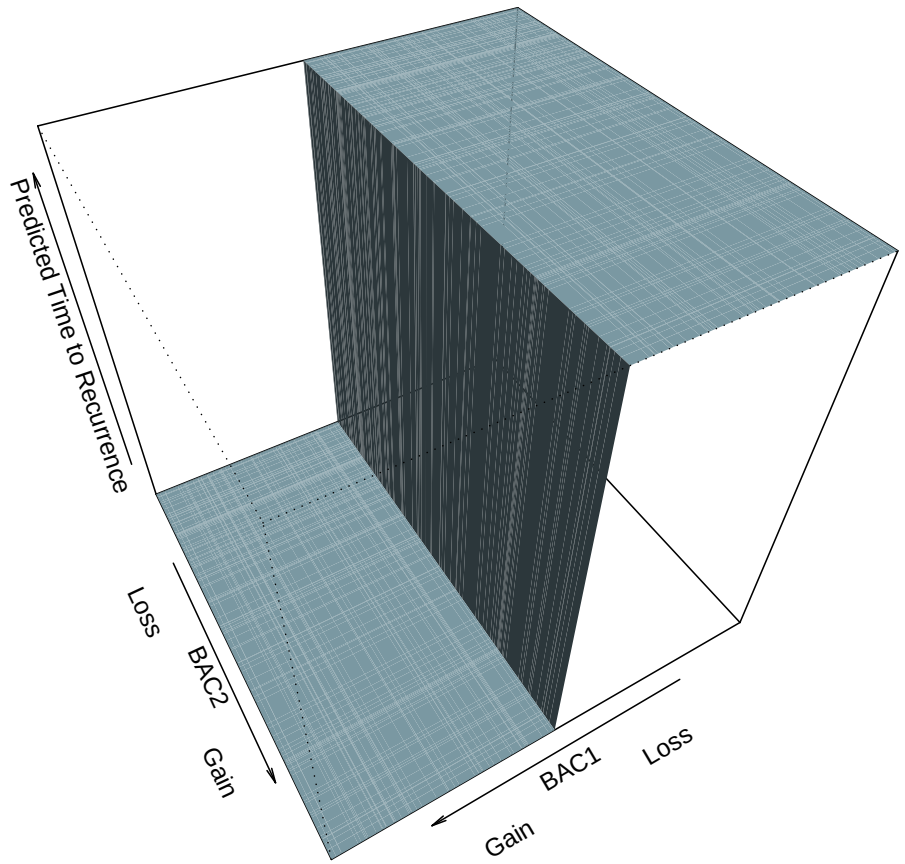
Sample			50 Repetitions				
Sample size	$v-fold$	Method	mean	std dev	avg size	sens vs. spec	ratio
250	2	dsacv	.0648	.002	2.88	35% 73%	1
		fscv	.0653	.001	2.74	26% 48%	.828
	5	dsacv	.0651	.002	3.58	39% 64%	1
		fscv	.0655	.001	2.34	23% 49%	.860
	10	dsacv	.0649	.002	3.88	41% 62%	1
		fscv	.0655	.001	2.30	23% 50%	.809
500	2	dsacv	.0642	.002	3.54	40% 69%	1
		fscv	.0649	.001	2.50	24% 49%	.698
	5	dsacv	.0640	.002	3.88	43% 64%	1
		fscv	.0650	.001	2.40	24% 50%	.606
	10	dsacv	.0639	.003	4.28	48% 63%	1
		fscv	.0650	.001	2.42	24% 50%	.567
1000	2	dsacv	.0636	.001	3.92	42% 61%	1
		fscv	.0645	.001	2.56	25% 50%	.537
	5	dsacv	.0637	.001	4.06	42% 58%	1
		fscv	.0645	.001	2.62	26% 50%	.574
	10	dsacv	.0636	.001	3.92	40% 59%	1
		fscv	.0647	.001	2.24	22% 50%	.503

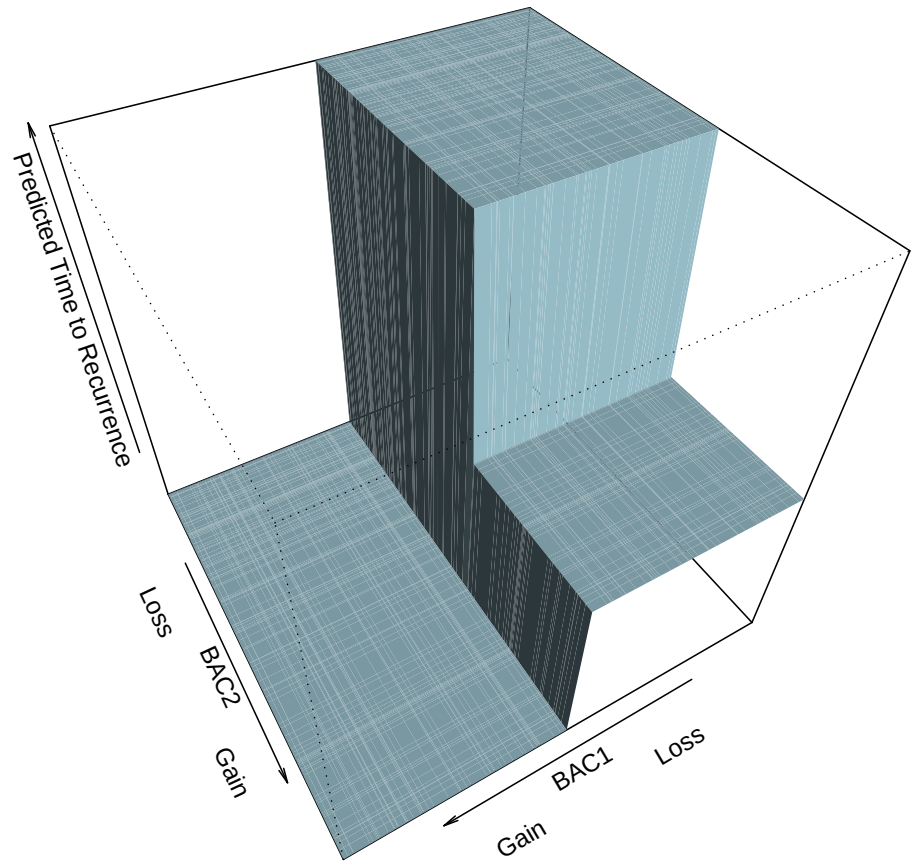
TREE REGRESSION

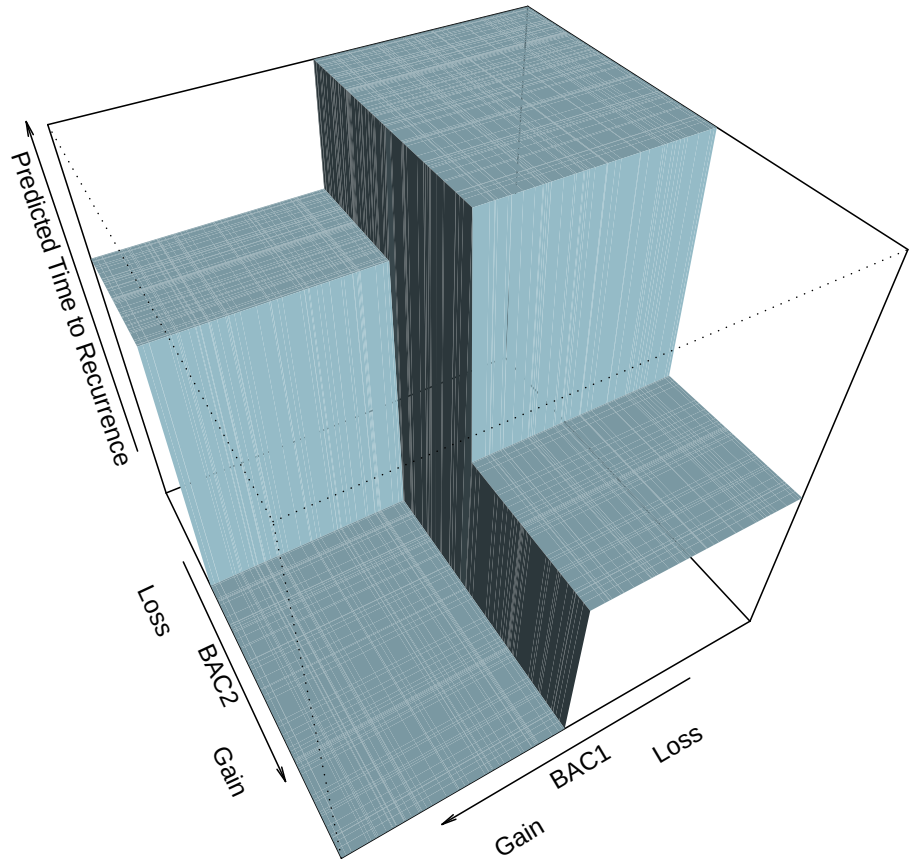


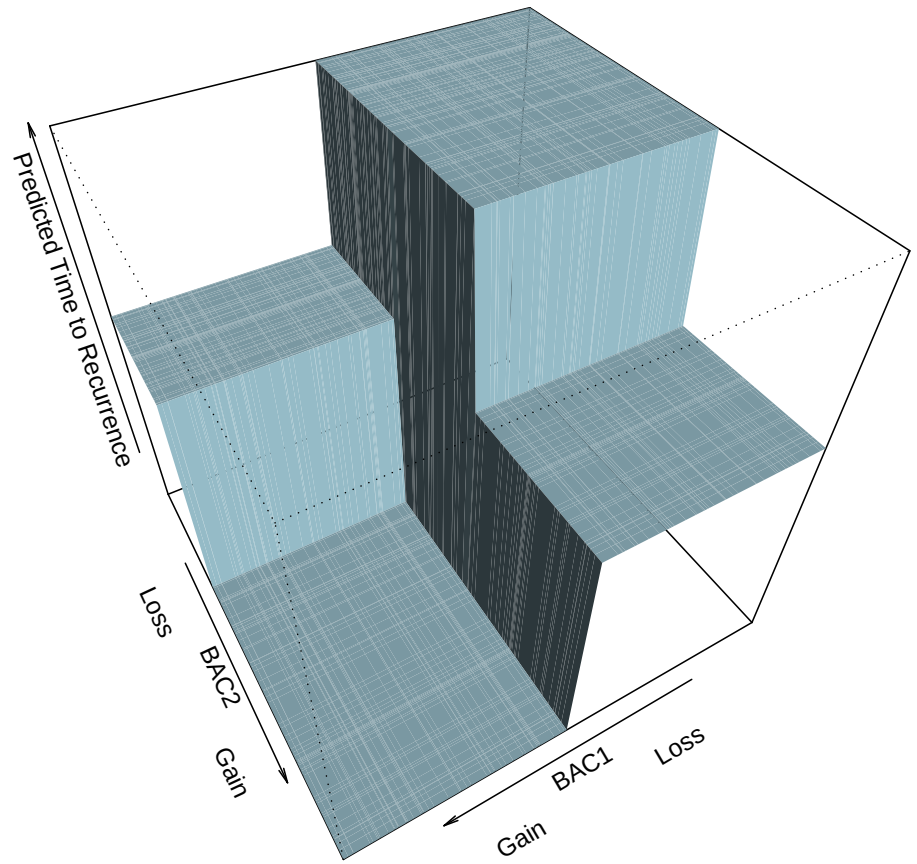
Chromosomal Location	
BAC 294	3q26
BAC 1226	10q22
BAC 529	5q11
BAC 542	5q21


Predicted Time to Recurrence
Loss
BAC2
Gain
Gain
BAC1
Loss


Predicted Time to Recurrence
Loss
BAC2
Gain
Gain
BAC1
Loss


Predicted Time to Recurrence
Loss
BAC2
Gain
Gain
BAC1
Loss


Predicted Time to Recurrence
Loss
BAC2
Gain
Gain
BAC1
Loss

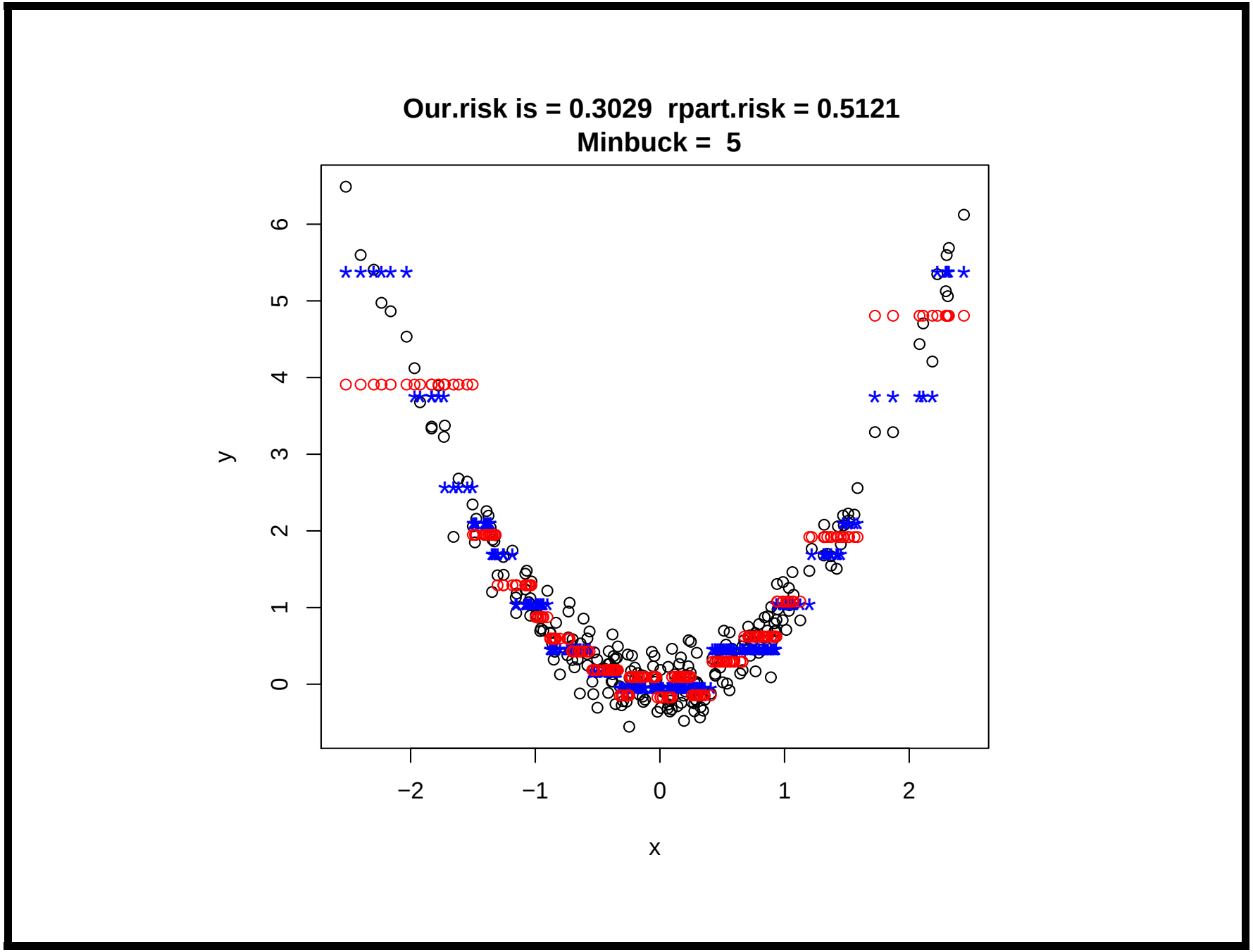

Predicted Time to Recurrence
Loss
BAC2
Gain
Gain
BAC1
Loss

Deletion/Substitution/Addition Algorithm in Partitioning

The goal of the D/S/A algorithm is to minimize a function $I \rightarrow f_{RSS}(I)$ over all allowed partitions (i.e., I) via three set functions:

- *Deletion* - Combine two regions to form a union resulting in $k-1$ basis functions
- *Substitution* - Form all possible combinations of two regions resulting in k basis functions
- *Addition* - Split one region into two distinct regions resulting in $k+1$ basis functions.

DSA versus CART in Univariate Histogram Regression


Our.risk is = 0.3029 rpart.risk = 0.5121
Minbuck = 5
y
x
0
1
2
3
4
5
6
−2
−1
0
1
2

Derivative-Based Importance Measures

- Given a particular b-specific fit $\hat{h}_b(W)$ for $b = 1, \ldots, B$, let
  $\bar{h}_{jb}(w) = \frac{1}{n} \sum_i \hat{h}_b(W_{1,i}, \ldots, W_{j-1,i}, w, W_{j+1,i}, \ldots, W_{d,i})$

- Estimate the importance measure for W_j:
  - Continuous
    $$\hat{\alpha}_b(j) = \frac{\int\limits_{w \in \mathcal{W}_j} |\frac{d}{dw} \bar{h}_{jb}(w)| dw}{\int\limits_{w \in \mathcal{W}_j} dw}$$
  - Binary
    $$\hat{\alpha}_b(j) = \mid \bar{h}_{jb}(1) - \bar{h}_{jb}(0) \mid$$

- The final estimate of the importance measure is then a weighted average of $\hat{\alpha}_b(j)$ across many b-specific fits:
  $$\hat{\alpha}(j) = \frac{\sum\limits_{b=1}^{B} \hat{\alpha}_b(j) I(j \in S_b) \mathrm{wt}_b}{\sum\limits_{b=1}^{B} I(j \in S_b) \mathrm{wt}_b}$$

Application in HIV-sequence analysis

Data Structure:

- 336 records linking the replication capacity (RC) with reverse transcriptase (RT) and protease (PRO) sequence data from individuals participating in studies at the San Francisco General Hospital and Gladstone Institute of Virology.
- The PRO positions 4-99 are RT positions 38-223 are used. In total there are 282 positions, with a median of 3 amino acids per positions.
- At each position there are a majority of patients exhibiting one amino acid as compared to the other possible amino acids at that position. There are 282 covariate positions, with the number of possible codons/position ranging from 1-11. We coded each position as a binary covariate.
- The outcome is a continuous measure of replication capacity.

ranging from 0.462 to 151.)

Approach:

- Select codon positions which are univariate significantly associated with HIV-replication, controlling the tail probability of the proportion of false positives at q (PFP(q)) with probability $1 - \alpha$.
- Run polynomial DSA-algorithm with the selected codons.
- Select fine-tuning parameters (e.g., also α) with cross-validation.

Controlling the Proportions of False Positives (PFP)

Given a multiple testing procedure $S_n(\alpha) \subset \{1, \ldots, 282\}$ controlling FWE at level α, one defines adjusted p-values as

$$\tilde{p}_{FWE}(j) = \inf\{\alpha \in [0,1] : j \in S_n(\alpha)\}.$$

Suppose null-hypotheses are already ordered by FWE-adjusted p-value. We proposed to control PFP by augmenting $S_n(\alpha)$ with the next $k(q)$ hypotheses, where $k(q)$ is defined by

$$\frac{k(q)}{\mid S_n(\alpha) \mid + k(q)} = q.$$

Adjusted p-values to control PFP Let

$U_n(\alpha) = \frac{1}{m}\sum_{l=1}^{m} I(\tilde{p}_{FWE}(l) \leq \alpha)$ Then, the adjusted p-values for controlling PFP(q) at level α (i.e., $P(V/R > q) \leq \alpha$) are given by:

$$p(j \mid PFP(q)) = U_n^{-1}\left((1-q)\frac{j}{m}\right), \quad j = 1....m.$$

Adjusted P-values: Controlling Proportion of False Positives

at levels q=0.01, 0.05, 0.10

q	rt215	rt41	pr54	pr90	rt184
0.01	0.0951	0.0440	0.0435	0.0404	0.00463
0.05	0.0841	0.0439	0.0433	0.0375	0.00442
0.10	0.0705	0.0438	0.0429	0.0338	0.00421

- By cross-validation, the complexity measure of the tensor products was chosen to be 2 (over various combinations of v-fold and max. terms), and the number of terms was chosen to be three. Therefore, final fit only contains 2-way interactions and three terms.
- Running the algorithm with:
  1. 10-fold; 5 maximum terms;
  2. 10-fold; 10 maximum terms;
  3. 5-fold; 5 maximum terms;

Resulted in the Same Model:

$$Y = 46.79 - 19.65(pr54) - 23.01(rt184) + 19.16(rt184) * (pr54).$$

Top 20 Codon Position Variable Importance Measures (VIM):

Codon Position	VIM
pr32	2.389988
pr47	2.480582
pr34	0.9502528
pr55	0.94073
pr90	0.6142873
rt184	0.6529078
pr54	0.4923045
pr43	0.5038835
rt41	0.3898231
pr46	0.4016424
pr82	0.2320981
rt215	0.2366182
rt121	0.3048936
pr10	0.1791156
pr71	0.1293414
rt135	0.1030811
rt102	0.1855826
rt67	0.1877864
rt83	0.1553970
pr18	0.3214662

Discussion of Results

rt184

- rt184 mutation is known from previous research, including in Segal et al. 2004, to be important in the replication capacity of a virus.
- A mutation in rt184 causes the virus to be unable to undergo compensatory mutagenesis to reestablish the wild type replication kinetics, and therefore full replication does not occur.
- **Note: Methionine (M) is the wild type amino acid of this position and Valine (V) is usually thought of as the drug resistant/mutant amino acid.

pr54

- This position is known to cause drug resistance (especially to Amprenavir (APV))
- The second most prevalent mutation pathway involves a mutation at pr54.
- pr54M or pr54L (Methionine or Leucine) cause an alteration in the shape of the protease inhibitor binding site, and therefore PIs cannot always bind (Hirsch 1998).
- pr54 V/L/T (Valine/Leucine/Threonine) increase resistance to each of the PIs when present with other mutations (Kempf 2000 and Hertogs 2000).

PREDICTING GENE EXPRESSION FROM SEQUENCE

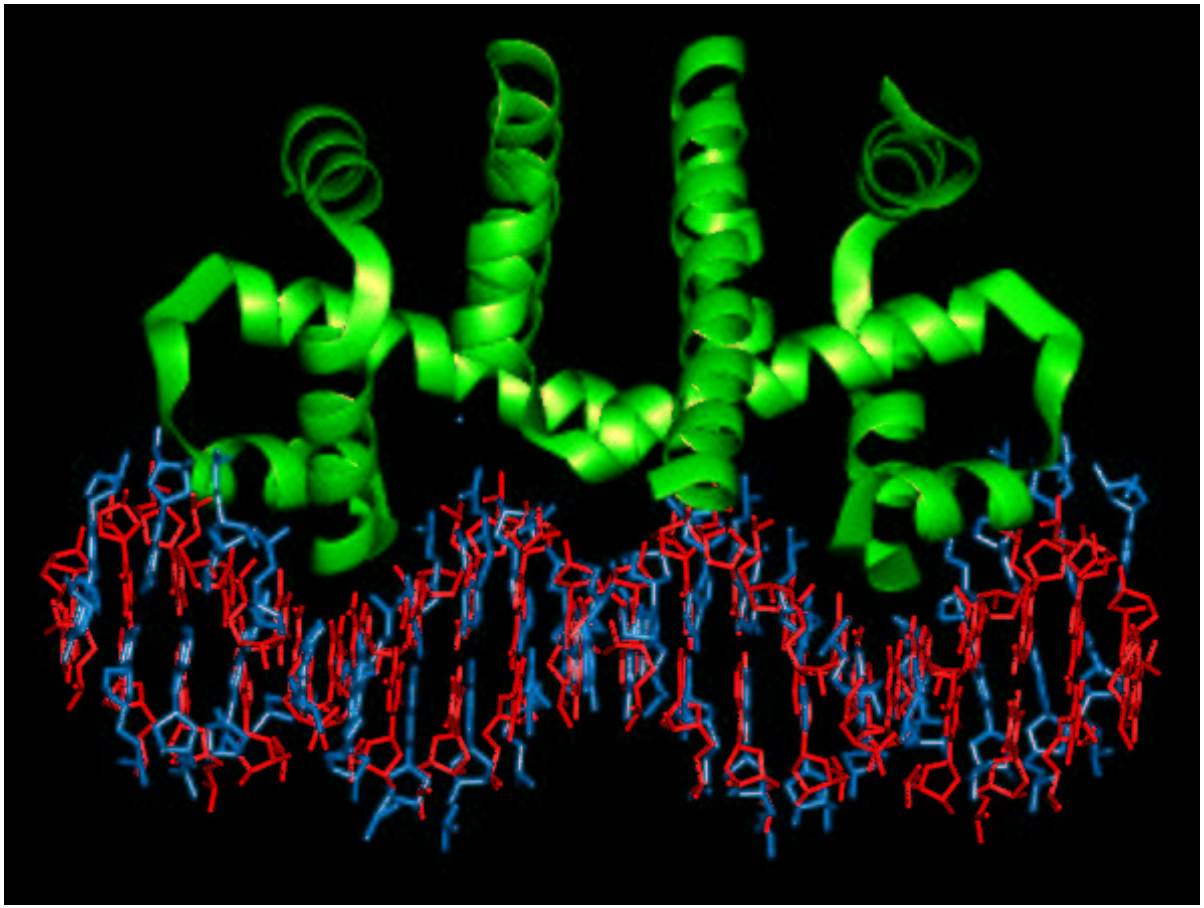
Goal: To identify binding sites (regulatory motifs).

n

Data:

- Gene Expression Data: $P \times N$ matrix with entries $Y_{ij}, i = 1, \cdots, P, j = 1, \cdots, N$. Y_{ij} is the logarithm of the relative gene expression for gene i in experiment j.
- Upstream Control Region (UCR): Roughly 600 to 1000 base pairs of the gene start site.

GAL4 BINDING



From `http://www.cryst.bbk.ac.uk/PPS2/.`

Transcription Factor Binding Site Identification

- *Identification of regulatory motifs in DNA sequences*
  Transcription factors (TF) are proteins that selectively bind to DNA to regulate gene expression. The transcription factor binding sites, or regulatory motifs, are short DNA sequences (5-25 base pairs) in the upstream control region (UCR) of genes, i.e., in regions roughly 600 to 1,000 base pairs from the gene start site (in lower eukaryotes, e.g., yeast).

Statistical question. Utilize gene expression data to identify sequence motifs associated with genes that are activated under a specified experimental condition.

Cell-cycle data for the yeast *Saccharomyces cerevisiae*

- 15 time points of the cell cycle
- Expression data for 2836 genes
- 512 distinct pairs of pentamers and reverse complements

  Explanatory variable: form sequence motif scores $\in [0, 1]$ of the number of occurrences of the given motif
- Model gene expression as a function of pentamers
- Use the D/S/A algorithm to extract most relevant pentamers:
  1. Select the number of tensor products s_1 via cross-validation
  2. Select s_1, s_2, and s_3 via cross-validation where $\max_{\vec{p} \in I} \sum_{j=1}^{d} I(p_j \neq 0) \leq s_2$ and $\max_{\vec{p} \in I} \sum_{j=1}^{d} p_j \leq s_3$
  3. Select s_0, s_1, s_2, and s_3 via cross-validation, where s_0 represents the dimension of the vector of covariates $\in \{1, \ldots, 512\}$
  4. Report importance measure for top ten pentamers

T=0 min
$(s_0 = 512,\ \hat{s}_1 = 5)$ $(AGGGG[stre])\ +\ (ACGCG[mcb])(CGAAA)$ $+\ (ATCCC)(CCTTA)(GCAAA)\ +\ (AAAAT)(CATCG)(GATGA)$ $+\ (ACCCG)(AGGGG[stre])(GAAAA[ecb])$
$(s_0 = 512,\ \hat{s}_1 = 5,\ \hat{s}_2 = 3,\ \hat{s}_3 = 3)$ $(AGGGG[stre])\ +\ (ACGCG[mcb])(CGAAA)$ $+\ (ATCCC)(CCTTA)(GCAAA)\ +\ (AAAAT)(CATCG)(GATGA)$ $+\ (ACCCG)(AGGGG[stre])(GAAAA[ecb])$
$(\hat{s}_0 = 55,\ \hat{s}_1 = 5,\ \hat{s}_2 = 3,\ \hat{s}_3 = 3)$ $(AGGGG[stre])\ +\ (ACGCG[mcb])$ $+\ (ATCCC)(CCTTA)\ +\ (CATCG)$ $+\ (ACCCG)(AGGGG[stre])(GAAAA[ecb])$

$T = 0$ min.	
Pentamer	VIM
AGGGG CCCCT [stre]	11.54
ACGCG CGCGT [mcb]	8.84
CATCG CGATG	8.11
ACCCC GGGGT	7.62
CGCGA TCGCG [scb]	7.43
GCCCC GGGGC	7.41
ATCCC GGGAT	7.24
AAGGG CCCTT	6.71
CCTTA TAAGG	5.23
GGGGA TCCCC	4.52

$T = 20$ min.	
Pentamer	VIM
ACGCG CGCGT [mcb]	21.55
CGCGA TCGCG [scb]	17.49
GACGC GCGTC	9.20
AGGGG CCCCT [stre]	7.19
AACGC GCGTT	5.24
AAGGG CCCTT	4.89
GCGAA TTCGC [scb]	2.43
GCGTA TACGC	2.33
CGAAA TTTCG [scb]	1.30
CACGA TCGTG	0.97

$T = 30$ min.	
Pentamer	VIM
ACGCG CGCGT [mcb]	10.51
CGCGA TCGCG [scb]	7.02
AACGC GCGTT	4.23
GACGC GCGTC	3.69
GCGTA TACGC	2.60
CTCCA TGGAG	2.15
CCACA TGTGG	2.07
CACAG CTGTG	1.63
GCGAA TTCGC [scb]	0.95
CCCAC GTGGG	0.02

Implementation: speed issues

- The D/S/A algorithm may require a substantial amount of fits: on a dataset with 1000 predictors, polynomial regression without constraints using 5-fold CV requires at least 5.000 OLS fits for a single substitution move – this may rise to as many as 100.000 as the size of the index set increases.
- Of cours, these numbers may be reduced by imposing constraints on the allowed moves.
- Nevertheless speed will be a major factor in any implementation of the algorithm.

Implementation: modularity

- A general implementation will allow the input of different loss functions, basis functions etc.
- But in a specific case one is usually able to deal better with the numerical problems by utilizing the structure of the specific problem.
- Hence a (good) implementation will allow the user to substitute several of the generic components, by a component designed for a specific case.

CONCLUDING REMARKS

- Loss Based Cross-validated DSA algorithms provide data adaptive algorithms (grounded by theory) for estimating parameters based on general data structures: Just determine loss function, and choose sieve.
- Optimal Censored data/Causal Inference data loss functions are obtained with the double robust IPCW/IPTW -full data loss function.
- Simulations suggest that the DSA-algorithm is asymptotically surprisingly (and fast) capable of truly minimizing the empirical risk function over all subsets of basis functions.
- In complex (i.e., genomic) studies we should let cross-validation make the choices: if we choose the parametrization/basis and corresponding subspaces with cross-validation, then the estimator becomes adaptive to the truth.

References

www.bepress.com/ucbbiostat/

- S. Keleş, M.J. van der Laan, and M.B. Eisen (2001). *Identification of Regulatory Elements Using A Feature Selection Method*. Division of Biostatistics, UC Berkeley, Technical Report No. 98. Bioinformatics.
- M.J. van der Laan and S. Dudoit (2003). *Unified Cross-Validation Methodology For Selection Among Estimators and a General Cross-Validated Adaptive Epsilon-Net Estimator: Finite Sample Oracle Inequalities and Examples*. Division of Biostatistics, UC Berkeley, Technical Report No. 130.
- A.M. Molinaro, S. Dudoit, and M.J. van der Laan (2003). *Tree-based Multivariate Regression and Density Estimation with Right-Censored Data*. Division of Biostatistics, UC Berkeley, Technical Report No. 135.
- S. Dudoit, M.J. van der Laan, S. Keleş, A.M. Molinaro, S.E. Sinisi, and S.L. Teng (2003). *Loss-Based Estimation with Cross-Validation: Applications to Microarray Data Analysis and Motif Finding*. Division of Biostatistics, UC Berkeley, Technical Report No. 137.
- van der Laan, M., Dudoit, S. Pollard, K. *Multiple Testing. Part III.*

Controlling the Proportion of False Positives and Generalized Family Wise Error 2004: UCB Division of Biostatictics Working Paper Series (paper 141).

- Dudoit, S., van der Laan, M., Pollard, K. *Multiple Testing. Part I. Single-Step Procedures for Control of General Type I Error Rates* 2003: UCB Division of Biostatictics Working Paper Series (paper 138).
- Pollard, K. and van der Laan, M. *Resampling -based Multiple Testing: Asymptotic Control of Type I Error and Application to Gene Expression Data* 2003: UCB Division of Biostatictics Working Paper Series (paper 121).

Possible Substitutions.