OPEN ACCESS Freely available online

PLOS ONE

LBoost: A Boosting Algorithm with Application for Epistasis Discovery

Bethany J. Wolf[1]*, Elizabeth G. Hill[1], Elizabeth H. Slate[2], Carola A. Neumann[3], Emily Kistner-Griffin[1]

1 Division of Biostatistics and Epidemiology, Medical University of South Carolina, Charleston, South Carolina, United States of America, **2** Department of Statistics, Florida State University, Tallahassee, Florida, United States of America, **3** Department of Cell and Molecular Pharmacology, Medical University of South Carolina, Charleston, South Carolina, United States of America

Abstract

Many human diseases are attributable to complex interactions among genetic and environmental factors. Statistical tools capable of modeling such complex interactions are necessary to improve identification of genetic factors that increase a patient's risk of disease. Logic Forest (LF), a bagging ensemble algorithm based on logic regression (LR), is able to discover interactions among binary variables predictive of response such as the biologic interactions that predispose individuals to disease. However, LF's ability to recover interactions degrades for more infrequently occurring interactions. A rare genetic interaction may occur if, for example, the interaction increases disease risk in a patient subpopulation that represents only a small proportion of the overall patient population. We present an alternative ensemble adaptation of LR based on boosting rather than bagging called LBoost. We compare the ability of LBoost and LF to identify variable interactions in simulation studies. Results indicate that LBoost is superior to LF for identifying genetic interactions associated with disease that are infrequent in the population. We apply LBoost to a subset of single nucleotide polymorphisms on the PRDX genes from the Cancer Genetic Markers of Susceptibility Breast Cancer Scan to investigate genetic risk for breast cancer. LBoost is publicly available on CRAN as part of the LogicForest package, http://cran.r-project.org/.

Citation: Wolf BJ, Hill EG, Slate EH, Neumann CA, Kistner-Griffin E (2012) LBoost: A Boosting Algorithm with Application for Epistasis Discovery. PLoS ONE 7(11): e47281. doi:10.1371/journal.pone.0047281

Editor: Jérémie Bourdon, Université de Nantes, France

Received December 19, 2011; **Accepted** September 14, 2012; **Published** November 8, 2012

Funding: This research was supported in part by pilot funding from an American Cancer Society Institutional Research Grant awarded to the Hollings Cancer Center, Medical University of South Carolina, by National Institutes of Health/National Institute of Dental and Craniofacial Research Grant K25DE016863, and by the South Carolina Clinical and Translational Research Institute, Medical University of South Carolina's CTSA, National Institutes of Health/National Center for Research Resources grant UL1RR029882. The contents are solely the responsibility of the authors and do not necessarily represent the official views of Natitional Institutes of Health. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: wolfb@musc.edu

Introduction

Many common diseases are heterogeneous, developing as a result of complex gene-gene and gene-environment interactions [1–3]. The heterogeneity of cancer, for example, is well documented and many authors note that distinct genetic patterns in cancer result in significant differences in disease outcome [4–6]. While a particular disease pathway may account for a majority of cases, there may be alternative pathways that account for only a small proportion of cases. Statistical methods capable of identifying key components in multiple disease pathways can aid in understanding an individual's risk of developing disease, in disease prognosis, and in prediction of response to therapy [7,8].

Logic regression (LR) is a single tree-based method capable of modeling high-order interactions [9]. LR generates classification rules by constructing Boolean (and = $\wedge$, or = $\vee$, and not = !) combinations of binary (0/1) predictors for classification of a binary response. For example, LR might produce the tree, $T = (x_4 \vee x_{11}) \wedge x_5 = (x_4 \wedge x_5) \vee (x_5 \wedge x_{11})$, which predicts a response value of 1 if either $x_4 \wedge x_5$ or $x_5 \wedge x_{11}$ are true. Otherwise, the predicted response is 0. All LR trees can be expressed as a disjunction of conjunctions as in the second expression for tree T. The conjunctive interactions described by the tree are referred to as *prime implicants* (PIs). Tree T is composed of the two PIs, $x_4 \wedge x_5$ and $x_5 \wedge x_{11}$, both of size 2 as each includes two variables. LR can identify PIs ranging in size from 1 to 8 predictors, and thus PI is a general term describing main effects and interactions. LR has been used in the development of screening and diagnostic tools for prostate and colorectal cancer, and to identify single nucleotide polymorphisms (SNPs) that confer risk in cardiovascular disease [10–13].

Tree-based classifiers are unbiased base classifiers but they are highly variable. The predictive accuracy of a tree-based classifier can be improved by using an ensemble of learners when predicting an observation's class [14–16]. The ensemble allows averaging across base learners resulting in an unbiased aggregated learner with reduced variability. One powerful approach to constructing ensemble-based learners is bagging, that is, the construction of classifiers from multiple bootstrap samples drawn from training data. Logic Forest (LF) is a bagged version of LR that generates an ensemble of logic regression-grown trees of varying sizes [17]. LF shows improved predictive performance over LR and is better able to discover PIs significantly associated with response, even in data with predictors measured with error and in data in which not all variables significantly associated with the response are observed. However, the ability of both LR and LF to recover PIs associated with response degrades for infrequently occurring PIs [17]. A rare PI would occur if, for example, the PI is highly predictive of

disease for a patient subpopulation that represents only a small proportion of the overall patient population.

Boosting is a powerful alternative algorithm for constructing ensemble learners that reweights the training data at successive iterations to improve prediction of observations poorly classified at previous iterations [18]. In this paper we present a boosted version of LR we refer to as LBoost, and introduce a measure of predictor importance. We compare the performance of LBoost relative to LF considering varying frequency of occurrence for PIs associated with response and varying model complexity. We also apply LBoost to a subset of SNP data from the Cancer Genetic Markers of Susceptibility (CGEMS) Breast Cancer Scan [19–21] to investigate genetic risk variants.

Methods

Define data $\mathbf{W}=(\mathbf{y},\mathbf{x})$ where $\mathbf{y}=(y_1,y_2,\ldots,y_n)$ is a vector of n binary responses and $\mathbf{x}$ is an $n\times p$ matrix of p binary predictors with $\mathbf{x}_i=(x_{i1},x_{i2},\ldots,x_{ip}),i=1,2,\ldots,n$. The algorithm for constructing an LBoost model is shown below.

LBoost Algorithm

For data set $\mathbf{W}$

1. Initialize a collection of observation-specific weights $\mathbf{w}_1$ where $\omega_{1j}=\frac{1}{n}$ and j indexes the number of observations, n.
2. For $a=1,2,...,A$ where A is the number of boosted LR trees constructed from data $\mathbf{W}$

a. Randomly select a positive integer $2\le M_a\le 8$ where M_a is the maximum number of predictors in an LR tree. (Random selection of tree size has been shown to modestly improve recovery of small PIs [17].)

b. Fit an LR tree, T_a, to data $\mathbf{W}$ using weights $\mathbf{w}_a$ and with no more than M_a predictors.

c. Compute the weighted error for T_a according to:

$$err_a=\frac{\sum_{j=1}^{n}\omega_{aj}I\left(y_j\neq\hat{y}_{aj}\right)}{\sum_{j=1}^{n}\omega_{aj}}$$

where $\hat{y}_{aj}$ is the predicted value for the jth observation from tree T_a

d. Using the weighted error compute a tree-specific weight for tree T_a according to:

$$\alpha_a=\ln\left(\frac{1-err_a}{err_a}\right)$$

e. Update observation-specific weights according to:

$$\omega_{a+1,j}=\omega_{aj}\exp\left(\alpha_a I\left(y_j\neq\hat{y}_{aj}\right)\right)$$

3. The forest of A boosted trees is $\mathrm{LB}(\mathbf{W},A)=\{T_1,T_2,\ldots,T_A\}=T_a$.

In step 2b, LBoost fits the LR tree using simulated annealing with misclassification error to choose between LR trees. Simulated annealing is the default search algorithm in LR. Use of misclassification for identifying the "best" LR model limits the number of trees fit at a given iteration of LBoost/LF to one tree with a maximum of 8 predictors.

We also use cross validation (CV) when constructing the forest for development of measures of model fit (Equation 2) and PI importance (Prime Implicant Importance Measures Section). For K-fold CV, let $\mathbf{W}_k=(\mathbf{y}_k,\mathbf{x}_k),k=1,2,\ldots,K$ be one of K approximately equally sized, non-overlapping subdivisions of the data. Given $\mathbf{W}_k$, let $\mathbf{W}_{-k}$ be the collection of all data subdivisions other than $\mathbf{W}_k$ such that $\mathbf{W}_{-k}=(\mathbf{W}_1,\ldots,\mathbf{W}_{k-1},\mathbf{W}_{k+1},\ldots,\mathbf{W}_K)$, and let n_k be the number of observations in $\mathbf{W}_{-k}$. We construct the kth LBoost model using $\mathbf{W}_{-k}$ according to the LBoost algorithm and use $\mathbf{W}_k$ as the kth test data set for the measures of model fit and PI importance. The final LBoost model therefore includes KA boosted trees and is denoted $\mathrm{LB}(\mathbf{W},KA)=\{T_{ka}\}$.

Now consider an observation $\mathbf{x}_i$ from the kth test data set $\mathbf{W}_k$. All trees within the boosted forest $\mathrm{LB}(\mathbf{W},KA)$ predict class membership for this observation. If predictor values in $\mathbf{x}_i$ produce a value of 1 for one or more of the PIs in tree T_{ka} within $\mathrm{LB}(\mathbf{W},KA)$, that tree predicts class membership $\hat{y}_{kai}(T_{ka})$ of 1; otherwise the tree predicts the class to be 0.

If we consider test data $\mathbf{W}_k$ as new data, we can make a CV prediction for the observations in $\mathbf{W}_k$ by taking a weighted average of the predictions for those trees in $\mathrm{LB}(\mathbf{W},KA)$ which were constructed from the corresponding training data $\mathbf{W}_{-k}$. We can use the test data set predictions to calculate an unbiased estimate of model error rate. For observation y_i in the test set corresponding to data W_{-k} (that is, for $y_i\in\mathbf{W}_k$), the boosted CV prediction from $\mathrm{LB}(\mathbf{W},KA)$ is

$$\hat{y}_i^{\mathrm{CV}}=\begin{cases}1 & \text{if}\quad \sum_{a=1}^{A}\alpha_{ka}(2\hat{y}_{kai}-1)I(y_i\in\mathbf{W}_k)\ge 0\\ 0 & \text{else.}\end{cases}\qquad(1)$$

Since predictions from a logic regression tree take values of either 0 or 1, the expression $(2\hat{y}_i-1)$ in equation 2 takes on values of 1 or -1, thereby allowing inclusion of all tree-specific weights α_{ka} in the final prediction. The CV misclassification rate for $\mathrm{LB}(\mathbf{W},KA)$ is

$$\mathrm{MC}^{\mathrm{CV}}=\frac{1}{n}\sum_{i=1}^{n}\left(y_i-\hat{y}_i^{\mathrm{CV}}\right)^2.\qquad(2)$$

Prime Implicant Importance Measure

In contrast to bagging, which applies the base learner to a bootstrap sample of the data, boosting is generally applied to the whole data set making it difficult to define an importance measure. To address this difficulty, we use CV to develop a measure of PI importance that can be estimated from an LBoost model, $\mathrm{LB}(\mathbf{W},KA)$. For tree T_{ka}, the CV misclassification rate for test data $\mathbf{W}_k$ is

$$\mathrm{MC.T}^{\mathrm{CV}}(T_{ka},\mathbf{y},\mathbf{x})=\frac{1}{n_k}\sum_{y_i\in W_k}I(y_i\neq\hat{y}_{kai}).\qquad(3)$$

Table 1. Two-locus interaction models.

Type 1	AA	Aa	aa	Type 2	AA	Aa	aa
BB	0	0	0	BB	0	0	0
Bb	0	1	1	Bb	0	0	0
bb	0	1	1	bb	0	0	1

Type 1 represents a DD interaction between SNPs a and b while Type 2 represents RR interaction between a and b. A value of 1 indicates SNP combinations conferring increased risk of disease.
doi:10.1371/journal.pone.0047281.t001

Let $\mathbf{x}_\ell$ be a PI occurring in tree T_{ka}, such that x_ℓ is an n-dimensional column vector of 0 s and 1 s corresponding to the PI's value for the n observations. We extract PIs from T_{ka} using the *prime.implicant* function available in the logicFS package [22]. Let $\mathbf{x}^{(\ell)}$ denote the matrix of all PIs in T_{ka} with $\mathbf{x}_\ell$ randomly permuted. Let $\text{MC.T}^{\text{CV}}_{\mathbf{x}_\ell}$ denote the tree-specific misclassification rate for T_{ka} applied to $\mathbf{x}^{(\ell)}$. The permutation based variable importance measure for $\mathbf{x}_\ell$ is defined by

$$\text{V.LB}(x_\ell) = \frac{1}{KA}\sum_{k=1}^{K}\sum_{a=1}^{A}\alpha_{ka}\left[\text{MC.T}^{\text{CV}}_{\mathbf{x}_\ell} - \text{MC.T}^{\text{CV}}\right]. \quad (4)$$

Simulation Studies

We conduct several simulations to examine the ability of LF and LBoost to recover PIs representing epistatic interactions between SNPs that are associated with disease. Two types of epistatic interactions are considered for the simulations comparing LBoost and LF (Table 1). An interaction of type 1 confers increased risk of disease when at least one copy of the minor allele is present from both loci; this type 1 interaction is referred to as the jointly dominant-dominant model (DD) [23–26]. An interaction of type 2 confers increased risk of disease if two copies of the minor allele are present from both loci; this type 2 interaction is referred to as the jointly recessive-recessive model (RR).

We consider three simulation scenarios: (1) the response is associated with a single DD interaction; (2) the response is associated with two DD interactions; and (3) the response is associated with a single RR interaction. We use the liability threshold model [27,28] to define all interaction models. Specifically, all simulated data are defined by the minor allele frequencies (MAFs) of the risk alleles, the disease prevalence, and the heritability of the epistatic interaction(s). For simplicity, risk alleles in an epistatic interaction have the same MAF. Also, for all simulations, the disease prevalence is set at 0.1 and the heritability for all epistatic interactions is set at 0.02. The disease prevalence was chosen to simulate a common disease such as breast cancer. The population level parameters for specific MAFs, threshold, and heritability are given in Table 2.

In addition to the SNPs in the epistatic interaction(s), additional non-causal SNPs are generated such that there are 100 SNPs in the final dataset. Minor allele frequencies for the non-causal SNPs are randomly selected from between 0.05 and 0.5. For simulation scenarios 1 and 2, all SNPs are coded as an indicator for at least one copy of the minor allele. For simulation scenario 3, SNPs are coded as the indicator for two copies of the minor allele. In scenarios 1 and 3 the response is associated with the DD or RR interaction between x_5 and x_{10}, thus the PI of interest is $x_5 \wedge x_{10}$. In scenario 2 the response is associated with two independent DD interactions, $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$.

We consider sample sizes ranging from 400 to 2400, generating 500 datasets for each simulation study. We examine the ability of LF and LBoost to recover the PIs known to be associated with the response using the variable importance measure for LF, V.LF [17], and V.LB for LBoost. Define F as the set of all PIs identified in either $\text{LF}(\mathbf{W},B)$ or $\text{LB}(\mathbf{W},KA)$. Let Q ($Q \subset F$) be the set of 20 PIs in $\text{LF}(\mathbf{W},B)$ or $\text{LB}(\mathbf{W},KA)$ with maximum absolute V.LF and V.LB (4) values, respectively. We say that the PI q, known to be associated with disease, has been recovered when $q \in Q$. We select the top 20 identified PIs because in the context of studying gene-gene interactions, 20 interactions represents $<1\%$ of all possible 2 locus combinations given 100 geneotyped SNPs.

We use the Logic Forest package in R v.2.14.1 [29] with simulated annealing optimization to fit all LF models [9,17]. For LBoost we use 5-fold CV and construct 20 trees for each dataset $\mathbf{W}_k$ resulting in an LBoost model with 100 LR trees. For comparisons, all LBoost and LF models include the same number of LR trees. The same starting and ending annealing temperatures are selected for LF and LBoost. The starting temperature of 2 is selected such that approximately 90% of "new" models are accepted. The final temperature of -1 is set to achieve a score where fewer than 5% of new models are accepted. The cooling schedule is set so that 50,000 iterations are required to get from start to end temperaure. Increasing the number of iterations to 250,000 does not affect our findings. With these settings, the LBoost algorithm constructs a model in less than a minute on a Windows 2.26 GHz machine.

Table 2. Population values for simulation parameters†.

Model‡	Minor Allele Frequency	Prob(PI+)	Prob(D+|PI+)	Prob(D+| PI−)	OR
	0.1	0.0361	0.2890	0.0930	3.961
Dominant-dominant	0.3	0.2601	0.1460	0.0839	1.866
	0.5	0.5625	0.1213	0.0726	1.763
	0.1	0.0001	1.0000	0.0975	Inf
Recessive-recessive	0.3	0.0081	0.6127	0.0955	14.98
	0.5	0.0625	0.2293	0.0915	2.952

†The disease prevalence is set at 0.1 and heritability is set a 0.02 for all simulations.
‡MAFs are the same for risk alleles in an epistatic interaction. Prob(PI+) is the probability that a subject has the PI. Prob(D+|PI+) and Prob(D+|PI−) are the probabilities a subject has disease given that they have the PI and do not have the PI respectively. OR is the population odds ratio given the model, MAF, and heritability.
doi:10.1371/journal.pone.0047281.t002

Results

Scenario 1: One Dominant-Dominant Interaction

n scenario 1, we investigate the ability of LBoost and LF to recover a single DD interaction that is associated with the response from among 100 binary variables. The minor allele frequencies of 0.1, 0.3, and 0.5 are considered. In data in which the MAFs for x_5 and x_{10} were 0.1, LBoost identified the combination $x_5 \wedge x_{10}$ more frequently than LF, although this difference was only significant for $n \geq 1200$ (Figure 1A). LBoost recovers $x_5 \wedge x_{10}$ in a maximum of 88.4% of simulations, while LF recovers this PI in a maximum of 81.0% of simulation runs. When the minor allele frequencies for x_5 and x_{10} are increased to 0.3, the ability of both LF and LBoost to recover $x_5 \wedge x_{10}$ improves. Under these conditions, LF recovers $x_5 \wedge x_{10}$ significantly more frequently than LBoost for $800 \leq n \leq 1600$ (Figure 1B). Both LF and LBoost recover the PI in $>75\%$ of simulation runs for $n \geq 1200$ and in more than 90% of simulation runs for $n \geq 1600$. In data in which the MAFs for x_5 and x_{10} are 0.5, LF and LBoost identify $x_5 \wedge x_{10}$ equally well, recovering this PI in $>80\%$ of simulation runs for $n \geq 1200$.

Scenario 2: Two Independent Dominant-Dominant Interactions

In the second scenario, we investigate the ability of LBoost and LF to recover 2 DD interactions that occur with different frequency. The MAFs for the two SNPs in the PI $x_{15} \wedge x_{20}$ are held constant at 0.1 while the MAFs for $x_5 \wedge x_{15}$ are set at 0.1, 0.3 or 0.5. In the first case, the MAFs for x_5,x_{10},x_{15}, *and* x_{20} are set at 0.1, thus the expected frequency of occurrence of the two PIs $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ are equivalent. For $n \leq 1600$, LF and LBoost recover the PIs equally well. However LF recovers both PIs significantly more frequently than LBoost for $n \geq 2000$ (see Figures 2A and 2B). LBoost recovers both PIs in $>70\%$ of simulation runs for $n = 2400$, however LF recovers both PIs $>80\%$ of simulation runs for the largest sample size.

In the second case, the MAFs for x_5 and x_{10} are increased to 0.3, but the frequencies of x_{15} and x_{20} are held at 0.1. In this case the PI $x_5 \wedge x_{10}$ occurs more frequently than $x_{15} \wedge x_{20}$. Both LF and LBoost recover $x_5 \wedge x_{10}$ more frequently than in the previous case. However, LF recovers this PI significantly more frequently than LBoost for $800 \leq n \leq 1600$ (Figure 2C). Both methods identify this PI in $>80\%$ of simulation runs for $n \geq 1200$. LBoost identifies the less frequently occurring PI, $x_{15} \wedge x_{20}$, significantly more often than LF for $n \geq 1200$ (Figure 2D).

In the third case, the MAFs for $x_5 \wedge x_{10}$ are increased to 0.5 holding the frequencies for x_{15} and x_{20} at 0.1. There is no significant difference in the proportion of times LF and LBoost recover $x_5 \wedge x_{10}$. Both methods recover this PI in $>80\%$ of simulation runs for $n \geq 1200$ (Figure 2E). However, LBoost recovers $x_{15} \wedge x_{20}$ significantly more frequently than LF for $n \geq 800$ (Figure 2F).

We also compare LBoost models with varying K-fold CV (K = 5, 10, and 20) with forest size KA = 100 for cases 1 and 3 for two independent DD interactions. In case 1 (MAF x_5,x_{10},x_{15}, *and* $x_{20} = 0.1$), LBoost identifies both PIs significantly more frequently using 5-fold CV relative to 20-fold CV for $n \geq 1600$ (Figure S1, panels A and B). However, there is not a significant difference between 5 and 10-fold CV. In case 3 (MAF x_5 and $x_{10} = 0.5$ and MAF x_{15} and $x_{20} = 0.1$), there is not a significant difference in the proportion of times LBoost recovers $x_5 \wedge x_{10}$ or $x_{15} \wedge x_{20}$ for 5, 10, and 20-fold CV at any sample size (Figure S1, panels C and D).

Additionally we examine the performance of LBoost in models with 100 (with 5-fold CV) and 200 (with 10-fold CV) trees holding the ratio of total number of trees, KA, to number of CV data set, K, constant at 20:1. Increasing the number of trees from 100 to 200 improves the proportion of times LBoost recovers the PIs in case 1 for $n \geq 1200$ though the difference is not significant (Figure S2, panels A and B). In case 3, there is no significant differences in the proportion of times LBoost recovers $x_5 \wedge x_{10}$ in models with 100 versus 200 trees (Figure S2, Panel C). However, LBoost identifies $x_{15} \wedge x_{20}$ significantly more often in models with 200 trees for $n \geq 1200$ though the difference is only significant for $n = 1600$.

Scenario 3: One Recessive-Recessive Interaction

In simulation scenario 3, we consider the ability of LF and LBoost to recover a single RR interaction. The probability of occurrence of the PI given disease status is less than in previous scenarios. As in the first scenario we consider MAFs of 0.1, 0.3, and 0.5 for x_5 and x_{10}. When both x_5 and x_{10} have MAFs of 0.1, the probability of observing $x_5 \wedge x_{10}$ given a subject is disease positive is 0.1%. This PI occurs so infrequently, neither LBoost nor LF identified $x_5 \wedge x_{10}$ at any sample size under consideration (results not shown).

When the MAFs of x_5 and x_{10} are increased to 0.3, the probability of the PI given a subject has disease increases to approximately 5%. In this case, LBoost identifies $x_5 \wedge x_{10}$ significantly more frequently than LF for $n \geq 1200$ (Figure 3, Panel A). LBoost identified this PI in a maximum of 67% of simulation runs, however LF identified it in a maximum of only 17% of simulation runs. When the minor allele frequencies for x_5 and x_{10} are increased to 0.5, the probability of $x_5 \wedge x_{10}$ given the subject has disease increases to 0.1433. The ability of both LF and LBoost to identify this PI is improved and both recover this PI in $>80\%$ of simulation runs for $n \geq 1600$ (Figure 3, Panel B). There is not a significant difference in the proportion of times each method recovers this PI at any sample size.

We also examine the performance of LF and LBoost in models with 100 (5-fold CV) and 200 (10-fold CV) trees holding the ratio of total number of trees, KA, to number of CV data set, K, constant at 20:1. Increasing the number of trees from 100 to 200 significantly improves the proportion of times LBoost recovers the $x_5 \wedge x_{10}$ for $n \geq 1200$ (Figure S3). In models with 200 trees, LBoost recovers $x_5 \wedge x_{10}$ in $>80\%$ of simulations for $n \geq 2000$ but recovers the PI in a maximum of 66.7% of simulations when LBoost models include 100 trees. Increasing the number of trees in a LF model does not significantly impact the ability of LF to recover $x_5 \wedge x_{10}$ (Figure S3).

Summary of Simulation Results

LF and LBoost exhibit similar ability to recover frequently occurring PIs. However, LBoost performs better than LF when PIs occur rarely (5 to 10% of the time among individuals with disease) and is better at recovering less frequent PIs in the presence of a frequently occurring PI. There is also a trend towards improved recovery of PIs with increasing the number of trees in an LBoost model regardless of frequency.

CGEMS Analysis

Peroxiredoxins (Prdxs) are a newly identified group of peroxidases upregulated in breast cancer [30–33]. No genetic analysis has been done so far to investigate the genomic integrity of the PRDX genes in breast or any other cancer. We investigate single nucleotide polymorphisms (SNPs) in the Cancer Genetic Markers of Susceptibility study (CGEMS) [19,20] data, available

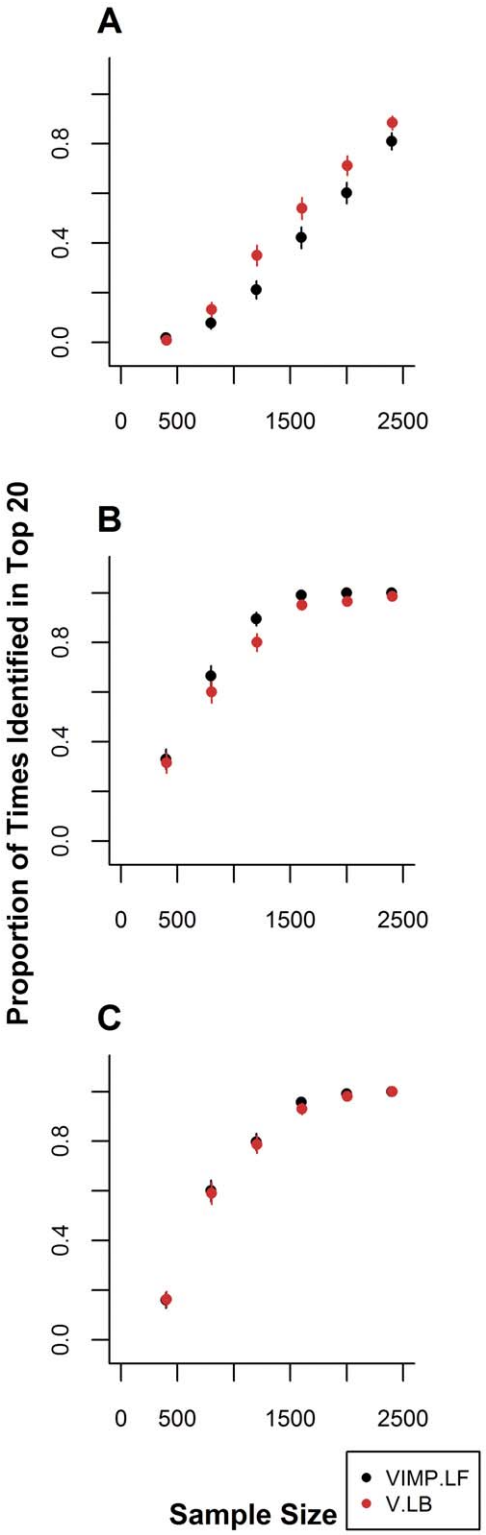

Figure 1. Recovery of the dominant-dominant interaction $x_5 \wedge x_{10}$ for MAFs of 0.1, 0.3, and 0.5. Each panel shows the proportion of times in 500 simulation runs the DD PI $x_5 \wedge x_{10}$ is recovered among the top 20 PIs by each method for different MAFs for x_5 and x_{10}. A) MAFs for x_5 and x_{10} are 0.1, panel B) MAFs for x_5 and x_{10} are 0.3, and panel C) MAFs for x_5 and x_{10} are 0.5. Error bars represent 95% confidence intervals.
doi:10.1371/journal.pone.0047281.g001

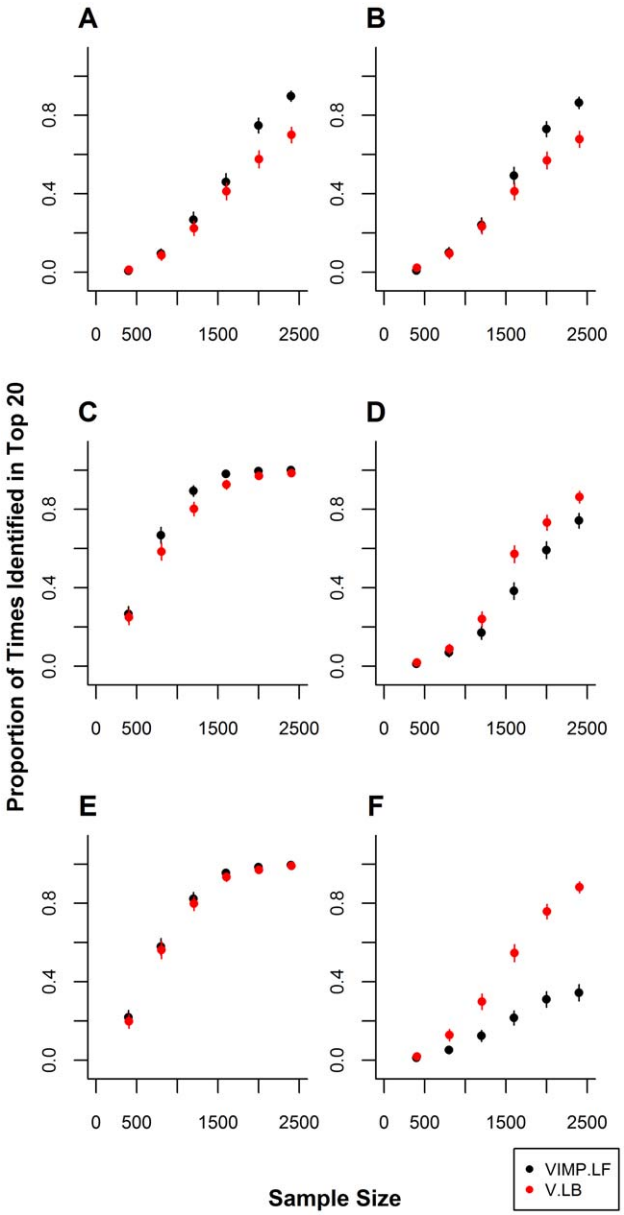

Figure 2. Recovery of the dominant-dominant interactions $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ for MAFs of 0.1, 0.3, and 0.5. Each panel shows the proportion of times in 500 simulation runs the DD PIs $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ are recovered among the top 20 PIs by each method for different MAFs. Specifically, Panels A) and B) show the proportion of times each method recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively when MAFs for x_5 and x_{10} are 0.1 and MAFs for x_{15} and x_{20} are 0.1. Panels C) and D) show the proportion of times each method recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively when MAFs for x_5 and x_{10} are 0.3 and MAFs for x_{15} and x_{20} are 0.1. Panels E) and F) show the proportion of times each methods recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively when MAFs for x_5 and x_{10} are 0.5 and MAFs for x_{15} and x_{20} are 0.1. Error bars represent 95% confidence intervals.
doi:10.1371/journal.pone.0047281.g002

in dbGaP (dbGaP accession number: ps000147.v1.p1). In total, 94 SNPs that are within 50 kb of the six PRDX genes are included in the LF and LBoost analyses.

The CGEMS study is an NCI-sponsored project begun in 2005 as a pilot study to identify genetic variants associated with increased risk of breast and prostate cancers. The CGEMS breast cancer data was derived from incident post-menopausal breast

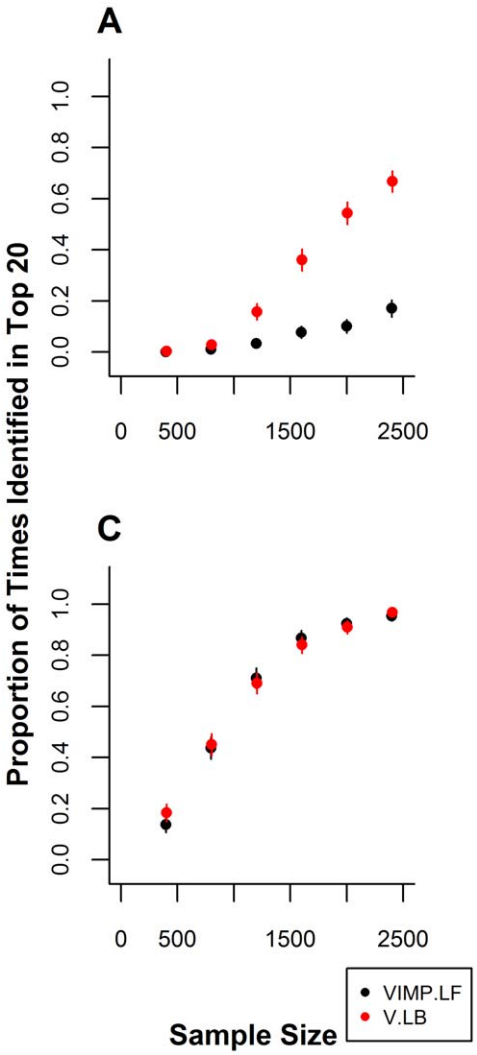

Figure 3. Recovery of the recessive-recessive interaction $x_5 \wedge x_{10}$ for MAFs 0.3 and 0.5. Each panel shows the proportion of times in 500 simulation runs the RR PI $x_5 \wedge x_{10}$ is recovered among the top 20 PIs by each method for different MAFs for x_5 and x_{10}. Panel A) MAFs for x_5 and x_{10} are 0.3 and panel B) MAFs for x_5 and x_{10} are 0.5. Error bars represent 95% confidence intervals.
doi:10.1371/journal.pone.0047281.g003

cancer cases in the Nurses' Health Study (NHS) arising between 1990 and 2004 [21]. Women in the CGEMS study provided a blood sample in 1989 or 1990 as part of the NHS and were cancer free at the time of sampling. In total 1145 incident cases were matched to 1142 controls from the NHS on age, blood collection time, ethnicity (all are self-reported Caucasian), and menopausal status at blood draw (all are menopausal at blood draw). Participants were genotyped using the Ilumina HumanHap550 chip. For each subject approximately 528,000 SNPs were genotyped providing coverage of 90% of the common SNPs.

Our analysis data comprised 94 SNPs on the six PRDX genes, coded for analysis by an indicator variable that takes value 1 if the subject has at least one copy of the minor allele in order to test the dominant effect of the minor allele. The LF and LBoost models constructed for these data both contain 100 trees. The LBoost model uses 5-fold cross-validation in model construction. The LF and LBoost models each identified over 300 unique PIs involving the 94 SNPs. PI importance was ranked from least to greatest according the VIMP.LF for the LF model and according to V.LB for LBoost. Empirical p-values were obtained for all PIs using a permutation approach.

Both LF and LBoost identified the PIs *rs11198819* ($p<0.01$) and *rs11198819* $\wedge$ *rs2297696* ($p<0.01$) among the top 5 most important PIs. The SNP *rs2297696* is upstream of PRDX3 on the sideroflexin 4 gene. The SNP *rs11198819* is downstream from PRDX3 in a non-coding region however, it is in strong linkage disequilibrium ($r^2=0.87$) with *rs3749562* which is on the PRDX3 gene. The remaining PIs in the LF model included *rs11198819* in conjunction with at least one additional SNP. LBoost identified two additional PIs not identified by LF, *rs1205171* ($p<0.025$) and *rs1205171* $\wedge$ *rs1461024* ($p<0.025$). The SNP *rs1205171* is found on the PRDX2 gene and *rs1461024* is found on the PRDX6 gene.

The moderate significance of these SNPs and SNP interactions is likely due to the fact that the PRDX family of genes does not play a dominant role in breast cancer. However, these results suggest possible associations of genetic variants within the PRDX family of genes with breast cancer. Additionally, LBoost identified SNPs not identified by LF. Further laboratory studies are necessary to explore the SNP interactions identified by LF and LBoost.

Discussion

Logic Forest, an ensemble adaptation of logic regression, has the ability to model complex interactions among binary predictors to describe disease state. However, LF is less adept in recovering rare PIs associated with disease, particularly in the presence of more frequent, predictive PIs. We introduced a boosting adaptation of LR referred to as LBoost in order to address this weakness of LF. Additionally we presented a predictor/PI importance measure based on permutation of a predictor or PI in the data, V.LB.

The results of the simulation study indicate that the ability of LF and LBoost to recover PIs associated with disease depends on the frequency with which a PI occurs in subjects that have disease and whether or not an additional predictive PI is present. In the scenario where the data only included a single DD interaction, LF and LBoost performed similarly, although LBoost showed modest improvement over LF in recovering $x_5 \wedge x_{10}$ when the minor allele frequency was low (0.1). In this case, the PI occurred in approximately 10% of subjects with disease. However, when the minor allele frequency increased to 0.3 (PI occurring in approximately 38% of subjects with disease) LF has better ability to recover the PI at smaller sample sizes. The greatest difference in ability to recover a single PI occurred in data where the interaction of interest was a recessive-recessive interaction in which the MAFs for x_5 and x_{10} were 0.3. In this case only 5% of subjects with disease were expected to have the PI $x_5 \wedge x_{10}$ and LBoost identified this PI significantly more often than LF for $n \geq 1200$.

In data with two interactions, LBoost recovered the less frequently occurring PI, $x_{15} \wedge x_{20}$ significantly more frequently than LF and performed similarly to LF in recovering the more frequently occurring PI. This difference in the ability to recover the rarer PI is more pronounced as the difference in frequency of occurrence between the two PIs increases.

For a fixed number of trees, increasing the number of CV sets, K, in an LBoost model moderately improves the ability of LBoost to identify frequent PIs. However, increasing the number of CV sets also decreases the ability LBoost to identify rare PIs. This effect is most pronounced in data with two or more PIs where both PIs are infrequent. However, the impact of varying the number of CV sets is small and choice of K and A should not greatly impact the ability of LBoost to identify PIs. From experience we have

found that selecting the total number of trees, KA, and the number of CV data sets, K, such that the ratio of total trees to number of CV data sets $10:1$ provides good balance for identifying frequent and rare PIs.

Increasing the total number of trees improves LBoost's ability to identify rare PIs. This effect is most noticeable when the PI is rare (i.e. the PI occurs in 5% of the cases), and is not evident for PIs that occur with greater than 10% frequency among cases. However, little additional computational time is necessary when increasing the forest size from 100 to 200 trees and therefore is advisable.

Both LBoost and LF are best suited for targeted investigation of SNP interactions associated with disease (e.g. pathway analysis). LBoost performs similarly to LF for frequently occurring PIs although LF performs better for mid-range sample sizes ($n=1200$ to 2000). However, LBoost is better able than LF to identify rare interactions that occur in approximately 5–10% of subjects with disease. LBoost is also better adapted to identify multiple PIs in situations where PI frequency varies among the PIs predictive of disease, a scenario more closely resembling a complex disease such as cancer. Since we can not know the data structure *a priori*, it is helpful to explore the predictor space using both methods.

Although we described the LBoost algorithm using LR with misclassification as the measure of goodness of fit, there are additional fit measures available in LR (e.g. deviance and least squares). There are also search algorithms other than simulated annealing that could be used to search for logical combinations of binary predictors. In subsequent work we will explore use of other LR measures of fit and additional search algorithms for identifying combinations of binary predictors in constructing LBoost models.

Supporting Information

Figure S1 Recovery of DD interactions $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ in LBoost models with 100 trees and 5, 10, or 20-fold CV. Each panel shows the proportion of times in 500 simulation runs the DD PIs $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ are recovered among the top 20 PIs by LBoost when the number of CV sets, K, is set to either 5, 10 or 20. The total number of LR trees in all models is held constant at $KA=100$. In all panels, black is LBoost with 5-fold CV, red is LBoost with 10-fold CV, and green is LBoost with 20-fold CV. Specifically, Panels A) and B) show the proportion of times LBoost recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively for different values of K when MAFs for x_5 and x_{10} are 0.1 and MAFs for x_{15} and x_{20} are 0.1. Panels C) and D) show the proportion of times LBoost recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively for different values of K when MAFs for x_5 and x_{10} are 0.5 and MAFs for x_{15} and x_{20} are 0.1. Error bars represent 95% confidence intervals.
(BMP)

Figure S2 Recovery of DD interactions $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ in LBoost models with 100 or 200 trees. Each panel shows the proportion of times in 500 simulation runs the DD PIs $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ are recovered among the top 20 PIs by LBoost when the number of LR trees in the LBoost model is either 100 or 200. We use 5-fold CV in LBoost models with 100 LR trees and 10-fold CV in models with 200 trees. Thus the ratio of total trees to k-fold CV is held constant at $20:1$. In all panels, black is LBoost with 100 trees and red is LBoost models with 200 trees. Specifically, Panels A) and B) show the proportion of times LBoost recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively for models with 100 and 200 trees when MAFs for x_5 and x_{10} are 0.1 and MAFs for x_{15} and x_{20} are 0.1. Panels C) and D) show the proportion of times LBoost recovers $x_5 \wedge x_{10}$ and $x_{15} \wedge x_{20}$ respectively for models with 100 and 200 trees when MAFs for x_5 and x_{10} are 0.5 and MAFs for x_{15} and x_{20} are 0.1. Error bars represent 95% confidence intervals.
(BMP)

Figure S3 Recovery of the RR interaction $x_5 \wedge x_{10}$ for MAF of 0.1 in LBoost models with 100 or 200 trees. The graph shows the proportion of times in 500 simulation runs the RR PI $x_5 \wedge x_{10}$ is recovered among the top 20 PIs by both when the number of LR trees in the LBoost or LF models is either 100 or 200. We use 5-fold CV in LBoost models with 100 LR trees and 10-fold CV in models with 200 trees. Thus the ratio of total trees to k-fold CV in all LBoost models is held constant at $20:1$. In all panels, black is LF models with 100 trees, red is LF models with 200 trees, green is LBoost models with 100 trees, and blue is LBoost models with 200 trees. Error bars represent 95% confidence intervals.
(BMP)

Acknowledgments

We appreciate the insightful comments of the reviewers which have significantly improved this manuscript.

Author Contributions

Conceived and designed the experiments: BW EH ES EK-G. Performed the experiments: BW EK-G. Analyzed the data: BW EK-G. Contributed reagents/materials/analysis tools: BW EK-G. Wrote the paper: BW EH ES CN EK-G. Obtained permission for use of the CGEMS data: EK-G.

References

1. Kumar S, Mohan A, Guleria R (2006) Biomarkers in cancer screening, research and detection: present and future: a review. Biomarkers 11: 385–405.
2. Alvarez-Castro JM, Carlborg O (2007) A unified model for functional and statistical epistasis and its application in quantitative trait loci analysis. Genetics 176: 1151–1167.
3. Kotti S, Bickeboller H, Clerget-Darpoux F (2007) Strategy for detecting susceptibility genes with weak or no marginal effects. Hum Hered 63: 85–92.
4. Pepe MS, Etzioni R, Feng Z, Potter JD, Thompson ML, et al. (2001) Phases of biomarker development for early detection of cancer. J Natl Cancer Inst 93: 1054–1061.
5. Srlie T, Perou CM, Tibshirani R, Aas T, Geisler S, et al. (2001) Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. Proc Natl Acad Sci U S A 98: 10869–10874.
6. Ertel A (2010) Bimodal gene expression and biomarker discovery. Cancer Inform 9: 11–14.
7. Kaklamani VG, Gradishar WJ (2006) Gene expression in breast cancer. Curr Treat Options Oncol 7: 123–128.
8. Baird AE (2010) Genetics and genomics of stroke: novel approaches. J Am Coll Cardiol 56: 245–253.
9. Ruczinski I, Kooperberg C, LeBlanc M (2003) Logic regression. J Comput Graph Stat 12: 475–511.
10. Etzioni R, Kooperberg C, Pepe M, Smith R, Gann PH (2003) Combining biomarkers to detect disease with application to prostate cancer. Biostatistics 4: 523–38.
11. Etzioni R, Falcon S, Gann PH, Kooperberg CL, Penson DF, et al. (2004) Prostate-specific antigen and free prostate-specific antigen in the early detection of prostate cancer: do combination tests improve detection? Cancer Epidem Biomark 13: 1640–5.
12. Janes H, Pepe M, Kooperberg C, Newcomb P (2005) Identifying target populations for screening or not screening using logic regression. Stats Med 24: 1321–38.
13. Kooperberg C, Bis JC, Marciante KD, Heckbert SR, Lumley T, et al. (2007) Logic regression for analysis of the association between genetic variation in the renin-angiotensin system and myocardial infarction or stroke. Amer J Epidemiol 165: 334–43.
14. Breiman L (1994) Bagging predictors. Technical Report 421, Department of Statistics, University of California at Berkley : 1–19.

15. Dietterich TG (2000) An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. Mach Learn 40: 139–57.
16. Friedman J (2001) Greedy function approximation: a gradient boosting machine. Annals Stat 29: 1189–1202.
17. Wolf BJ, Hill EG, Slate EH (2010) Logic forest: an ensemble classifier for discovering logical combinations of binary markers. Bioinformatics 26: 2183–2189.
18. Freund SR Y (1997) A decision-theoretic generalization of online learning and an application to boosting. J Comput Sys Sci 55: 119–139.
19. National Cancer Institute (2005). Cancer genetic markers of susceptibility (cgems). Available: http://cgems.cancer.gov/data/. Accessed 2010 October.
20. Hunter DJ, Kraft P, Jacobs KB, Cox DG, Yeager M, et al. (2007) A genome-wide association study identifies alleles in FGFR2 associated with risk of sporadic postmenopausal breast cancer. Nat Genet 39: 870–874.
21. Tworoger SS, Eliassen AH, Sluss P, Hankinson SE (2007) A prospective study of plasma prolactin concentrations and risk of premenopausal and postmenopausal breast cancer. J Clin Oncol 25: 1482–1488.
22. Schwender H (2007) logicFS: Identifying interesting SNP interactions with logicFS. Bioconductor package.
23. Li W, Reich J (2000) A complete enumeration and classification of two-locus disease models. Hum Hered 50: 334–349.
24. Hallgrmsdttir IB, Yuster DS (2008) A complete classification of epistatic two-locus models. BMC Genet 9: 17.
25. Cordell HJ (2002) Epistasis: what it means, what it doesn't mean, and statistical methods to detect it in humans. Hum Mol Genet 11: 2463–2468.
26. VanderWeele TJ, Laird NM (2011) Tests for compositional epistasis under single interactionparameter models. Ann Hum Genet 75: 146–156.
27. Dempster ER, Lerner IM (1950) Heritability of threshold characters. Genetics 35: 212–236.
28. Wray NR, Goddard ME (2010) Multi-locus models of genetic risk of disease. Genome Med 2: 10.
29. R Development Core Team (2009). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. Available: http://www.R-project.org. URL http://www.R-project.org. Accessed 2012 March.
30. Bae JY, Ahn SJ, Han W, Noh DY (2007) Peroxiredoxin I and II inhibit h2o2-induced cell death in mcf-7 cell lines. J Cell Biochem 101: 1038–1045.
31. Cao J, Schulte J, Knight A, Leslie NR, Zagozdzon A, et al. (2009) Prdx1 inhibits tumorigenesis via regulating pten/akt activity. EMBO J 28: 1505–1517.
32. Noh DY, Ahn SJ, Lee RA, Kim SW, Park IA, et al. (2001) Overexpression of peroxiredoxin in human breast cancer. Anticancer Res 21: 2085–2090.
33. Wang T, Tamae D, LeBon T, Shively JE, Yen Y, et al. (2005) The role of peroxiredoxin II in radiation-resistant MCF-7 breast cancer cells. Cancer Res 65: 10338–10346.