Contrast Trees and Distribution Boosting[*]

Jerome H. Friedman[†]

Abstract

A new method for decision tree induction is presented. Given a set of predictor variables $\mathbf{x} = (x_1, x_2, \cdots, x_p)$ and *two* outcome variables y and z associated with each $\mathbf{x}$, the goal is to identify those values of $\mathbf{x}$ for which the respective distributions of $y \,|\, \mathbf{x}$ and $z \,|\, \mathbf{x}$, or selected properties of those distributions such as means or quantiles, are most different. Contrast trees provide a lack-of-fit measure for statistical models of such statistics, or for the complete conditional distribution $p_y(y \,|\, \mathbf{x})$, as a function of $\mathbf{x}$. They are easily interpreted and can be used as diagnostic tools to reveal and then understand the inaccuracies of models produced by any learning method. A corresponding contrast boosting strategy is described for remedying any uncovered errors thereby producing potentially more accurate predictions. This leads to a distribution boosting strategy for directly estimating the full conditional distribution of y at each $\mathbf{x}$ under no assumptions concerning its shape, form or parametric representation.

Keywords: prediction diagnostics, classification, regression, boosting, quantile regression, conditional distribution estimation

Significance

Often machine learning methods are applied and results reported in cases where there is little to no information concerning accuracy of the output. Simply because a computer program returns a result does not insure its validity. If decisions are to be made based on such results it is important to have some notion of their veracity. Contrast trees represent a new approach for assessing the accuracy of many types of machine learning estimates that are not amenable to standard validation methods. In situations where inaccuracies are detected boosted contrast trees can often improve performance. A special case, distribution boosting, provides an assumption free method for estimating the full probability distribution of an outcome variable given any set of joint input predictor variable values.

1 Introduction

In statistical (machine) learning one has a system under study with associated attributes or variables. The goal is to estimate the unknown value of one of the variables y, given the known joint values of other (predictor) variables $\mathbf{x} = (x_1 \cdots, x_p)$ associated with the system. It is seldom the case that a particular set of $\mathbf{x}$–values gives rise to a unique value for y. There are quantities other than those in $\mathbf{x}$ that influence y whose values are neither controlled nor observed. Specifying a particular set of joint values for $\mathbf{x}$ results in a probability distribution of possible y-values, $p_y(y \,|\, \mathbf{x})$, induced by the varying values of the uncontrolled quantities. Given a sample $\{y_i, \mathbf{x}_i\}_{i=1}^N$ of previous solved cases, the goal is to estimate the distribution $p_y(y \,|\, \mathbf{x})$, or some of its properties, as a function of the predictor variables $\mathbf{x}$. These can then be used to predict likely values of y realized at each $\mathbf{x}$.

Usually only a single property of $p_y(y \,|\, \mathbf{x})$ is used for prediction, namely a measure of its central tendency such as the mean or median. This provides no information concerning prediction accuracy at each $\mathbf{x}$. Only collective accuracy over a set of $\mathbf{x}$-values can be estimated using cross-validation. In order to estimate individual prediction accuracy at each $\mathbf{x}$ one needs additional properties of $p_y(y \,|\, \mathbf{x})$ such as various quantiles, or the distribution itself. These can be estimated as functions of $\mathbf{x}$ using maximum likelihood or minimum risk techniques. Such methods however do not provide a measure of accuracy (goodness-of-fit) for their respective estimates as functions of $\mathbf{x}$. There is no way to know how well the results actually characterize the distribution of y at each $\mathbf{x}$.

Contrast trees can be used to assess lack-of-fit of any estimate of $p_y(y \,|\, \mathbf{x})$, or its properties (mean, quantiles), as a function of $\mathbf{x}$. In cases where the fit is found to be lacking, contrast *boosting* applied to the output can often improve accuracy. A special case of contrast boosting, *distribution* boosting, can be used to estimate the full conditional distribution $p_y(y \,|\, \mathbf{x})$ under no assumptions. Contrast trees can also be used to uncover concept drift and reveal discrepancies in the predictions of different learning algorithms.

2 Contrast trees

Contrast trees are close cousins of regression trees (Breiman *et al* 1984). A regression tree partitions the space of $\mathbf{x}$ - values into easily interpretable regions defined by simple conjunctive

[*] Classification: Physical Sciences (Statistics)

[†] Department of Statistics, Stanford University, Stanford, 94305, USA. (jhf@stanford.edu)

rules. The goal is to produce regions in $\mathbf{x}$ - space such that the variation of y values within each is made small. A *contrast* tree also partitions the $\mathbf{x}$ - space into similarly defined regions, but with a different purpose. There are *two* outcome variables y and z associated with each $\mathbf{x}$. The goal is to find regions in $\mathbf{x}$ - space where the values of the two variables are most different.

In some applications of contrast trees the outcomes y and z can be different functions of $\mathbf{x}$, $y = f(\mathbf{x})$ and $z = g(\mathbf{x})$, such as predictions produced by two different learning algorithms. The goal of the contrast tree is then to identify regions in $\mathbf{x}$ - space where the two predictions most differ. In other cases the outcome y may be observations of a random variable assumed to be drawn from some distribution at $\mathbf{x}$, $y \sim p_y(y \,|\, \mathbf{x})$. The quantity z might be an estimate for some property of that distribution such as its estimated mean $\hat{E}(y \,|\, \mathbf{x})$ or p-th quantile $\hat{Q}_p(y \,|\, \mathbf{x})$ as a function of $\mathbf{x}$. One would like to identify $\mathbf{x}$ - regions where the estimates z appear to be the least compatible with the actual empirical distribution of y. Alternatively z itself could be a random variable independent of y (given $\mathbf{x}$) with distribution $p_z(z \,|\, \mathbf{x})$ and interest is in identifying regions of $\mathbf{x}$ - space where the two distributions $p_y(y \,|\, \mathbf{x})$ and $p_z(z \,|\, \mathbf{x})$ most differ.

In these applications contrast trees can be used as diagnostics to ascertain the lack-of-fit of statistical models to data or to other models. As with other tree based methods the uncovered regions are defined by conjunctive rules based on simple logical statements concerning the variable values. Thus it is straightforward to understand the joint predictor variable values at which discrepancies have been identified. Such information may temper confidence in some predictions or suggest ways to improve accuracy.

In prediction problems z is taken to be an estimate of some property of the distribution $p_y(y \,|\, \mathbf{x})$, or of the distribution itself. One way to improve accuracy is to modify the predicted values z in a way that reduces their discrepancy with the actual values as represented by the data. Contrast trees attempt to identify regions of $\mathbf{x}$ - space with the largest discrepancies. The z - values within in each such region can then be modified to reduce discrepancy with y. This produces new values of z which can then be contrasted with y using another contrast tree. This process can then be applied to the regions of the new tree thereby producing further modified z - values. This "boosting" strategy of successively building contrast trees on the output of previously induced trees can be continued until the average discrepancy stops improving.

3 Building contrast trees

The data consists of N observations $\{\mathbf{x}_i, y_i, z_i\}_{i=1}^N$ each with a joint set of predictor variable values $\mathbf{x}_i$ and two outcome variables y_i and z_i. Contrast trees are constructed from this data in an iterative manner. At the Mth iteration the tree partitions the space of $\mathbf{x}$ - values into M disjoint regions $\{R_m\}_{m=1}^M$ each containing a subset of the data $\{\mathbf{x}_i, y_i, z_i\}_{\mathbf{x}_i \in R_m}$. At the first iteration there is a single region containing the entire data set. Associated with any data subset is a discrepancy measure between the y and z values of the subset

$$d_m = D(\{y_i\}_{\mathbf{x}_i \in R_m}, \{z_i\}_{\mathbf{x}_i \in R_m}). \tag{1}$$

Choice of a particular discrepancy measure depends on the specific application as discussed in Section 4.

At the next $(M+1)$st iteration each of the regions R_m defined at the Mth iteration $(1 \leq m \leq M)$ is provisionally partitioned (split) into two regions $R_m^{(l)}$ and $R_m^{(r)}$ ($R_m^{(l)} \cup R_m^{(r)} = R_m$). Each of these "daughter" regions contains its own data subset with associated discrepancy measure $d_m^{(l)}$ and $d_m^{(r)}$ (1).

Within each separate region the quality of a split is defined as the product of two factors

$$Q_m(l,r) = (f_m^{(l)} \cdot f_m^{(r)}) \cdot \max(d_m^{(l)}, d_m^{(r)})^\beta. \tag{2}$$

In the first, $f_m^{(l)}$ and $f_m^{(r)}$ are the fraction of observations in the "parent" region R_m associated with each of the two daughters. This factor discourages highly asymmetric splits in anticipation of further splitting. The second factor attempts to isolate daughter regions with high discrepancy. The parameter β regulates the relative influence of the two factors. Results are insensitive to its value. In all examples below the default $\beta = 2$ was used.

The types of splits considered here are the same as in ordinary regression trees (Breiman *et al* 1984). Each involves one of the predictor variables x_j. For numeric variables splits are specified by a particular value of that variable (split point) s. The corresponding daughter regions are defined by

$$\begin{aligned} \mathbf{x} \in R_m \,\&\, x_j \leq s &\Longrightarrow \mathbf{x} \in R_m^{(l)} \\ \mathbf{x} \in R_m \,\&\, x_j > s &\Longrightarrow \mathbf{x} \in R_m^{(r)}. \end{aligned} \tag{3}$$

For categorical variables (factors) the respective levels are ordered by discrepancy (1). The discrepancy at each respective level of the factor for the observations in the mth region is computed. Splits are then considered in this order.

Within each current region R_m all possible splits are performed and the one maximizing (2) is associated with that region. Then the region whose associated split maximizes actual improvement

$$I_m = \max(d_m^{(l)}, d_m^{(r)}) - d_m \tag{4}$$

is ultimately chosen to create the two new regions at that iteration. These new regions replace the corresponding parent producing $M+1$ total regions. Splitting stops when no estimated improvement (4) is greater than zero, the tree reaches a specified size or the observation count within all regions is below a specified threshold.

Tree size (number of regions) is generally specified by the user. It involves a trade-off between discrepancy and interpretability. Smaller trees give rise to larger regions defined by simpler conjunctive rules and are thereby easier to interpret. Larger trees have the potential to uncover smaller regions of higher discrepancy defined by more complex rules. Pruning strategies analogous to those in CART (Breiman *et al* 1984) based on cross-validation can also be employed to guide choice of tree size.

4 Discrepancy measures

By defining different discrepancy measures contrast trees can be applied to a variety of different problems. Even within a particular type of problem there may be a number of different appropriate discrepancy measures that can be used.

When the two outcomes are simply functions of $\mathbf{x}$, $y = f(\mathbf{x})$ and $z = g(\mathbf{x})$, any quantity that reflects their difference in values at the same $\mathbf{x}$ can be used to form a discrepancy measure such as

$$d_m = \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m} |\, y_i - z_i \,|. \tag{5}$$

Here N_m is the number of observations in the region R_m. If y is a random variable and z is an estimate for the mean of its conditional distribution at $\mathbf{x}$, $z_i = \hat{E}(y \,|\, \mathbf{x}_i)$, a natural discrepancy measure is

$$d_m = \frac{1}{N_m} \left| \sum_{\mathbf{x}_i \in R_m} (y_i - z_i) \right|. \tag{6}$$

This discrepancy (6) reflects the absolute difference between the empirical mean of the outcomes $\{y_i\}_{\mathbf{x}_i \in R_m}$ and that of the corresponding predictions $\{z_i\}_{\mathbf{x}_i \in R_m}$ in the region. Alternatively, if z is an estimate for the pth quantile at $\mathbf{x}$, $z_i = \hat{Q}_p(y \,|\, \mathbf{x}_i)$, a natural discrepancy measure would be lack-of-coverage

$$d_m = \left| p - \frac{1}{N_m} \sum_{\mathbf{x}_i \in R_m} I(y_i < z_i) \right|. \tag{7}$$

If $y \sim p_y(y \,|\, \mathbf{x})$ and $z \sim p_z(z \,|\, \mathbf{x})$ are both independent random variables associated with each $\mathbf{x}$, a discrepancy measure reflects the distance between their respective distributions. There are many proposed empirical measures of distribution distance. Every two–sample test has one. For the examples below a variant of the Anderson–Darling (Anderson and Darling 1952) statistic is used. Let $\{t_i\} = \{y_i\} \cup \{z_i\}$ represent the pooled (y, z) sample in a region R_m. Then discrepancy between the distributions of y and z is taken to be

$$d_m = \frac{1}{2N_m - 1} \sum_{i=1}^{2N_m - 1} \frac{\left| \hat{F}_y(t_{(i)}) - \hat{F}_z(t_{(i)}) \right|}{\sqrt{i \cdot (2N_m - i)}} \tag{8}$$

where $t_{(i)}$ is the ith value of t in sorted order, and $\hat{F}_y$ and $\hat{F}_z$ are the respective empirical cumulative distributions of y and z. Note that this discrepancy measure (8) can be employed in the presence of arbitrarily censored or truncated data simply by employing a nonparametric method to estimate the respective CDF's such as Turnbull (1976) .

Discrepancy measures can be, and often are, customized to particular applications. In this sense they are similar to loss criteria in prediction problems. However, in the context of contrast trees (and boosting) there is no requirement that they be convex or even differentiable. Moreover, discrepancies need not be expressible as a sum of terms each involving a single observation as in (5). Examples are (6) (7) (8).

5 Boosting contrast trees

As indicated above, and illustrated in the examples presented below and in the Supporting Information, contrast trees can be employed as diagnostics to examine the lack of accuracy of predictive models. To the extent that inaccuracies are uncovered, *boosted* contrast trees can be used to attempt to mitigate them, thereby producing more accurate predictions. Contrast boosting derives successive modifications to an initially specified z, each reducing its discrepancy with y over the data. Prediction then involves starting with the initial value of z and then applying the modifications to produce the resulting estimate.

5.1 Estimation contrast boosting

In this case z is taken to be an estimate of some parameter of $p_y(y \,|\, \mathbf{x})$. The z - values within each region $R_m^{(1)}$ of a contrast tree can be modified $z \to z^{(1)} = z + \delta_m^{(1)}$ $(\mathbf{x} \in R_m^{(1)})$ so that the discrepancy (1) with y is zero in that region

$$D(\{y_i\}_{\mathbf{x}_i \in R_m^{(1)}}, \{z_i^{(1)}\}_{\mathbf{x}_i \in R_m^{(1)}}) = 0. \tag{9}$$

This in turn yields zero average discrepancy between y and $z^{(1)}$ over the regions defined by the terminal nodes of the corresponding contrast tree. However, there may well be other partitions of the $\mathbf{x}$ - space defining different regions $\{R_m^{(2)}\}_1^M$ for which this discrepancy is not small. These may be uncovered by building a second tree to contrast y with $z^{(1)}$ producing updates

$$z^{(2)} = z^{(1)} + \delta_m^{(2)} \; (\mathbf{x} \in R_m^{(2)}). \tag{10}$$

These in turn can be contrasted with y to produce new regions $\{R_m^{(3)}\}_1^M$ and corresponding updates $\{\delta_m^{(3)}\}_1^M$. Such iterations can be continued K times until the updates become small. As with gradient boosting (Friedman 2001) performance accuracy is often improved by imposing a learning rate. At each step k the computed update $\delta_m^{(k)}$ in each region $R_m^{(k)}$ is reduced by a factor $0 < \alpha \leq 1$. That is, $\delta_m^{(k)} \leftarrow \alpha\, \delta_m^{(k)}$ in (10).

Each tree k in the boosted sequence $1 \leq k \leq K$ partitions the $\mathbf{x}$ - space into a set of regions $\{R_m^{(k)}\}$. Any point $\mathbf{x}$ lies within one region $m_k(\mathbf{x})$ of each tree with corresponding update $\delta^{(k)}_{m_k(\mathbf{x})}$. Starting with a specified initial value $z(\mathbf{x})$ the estimate $\hat{z}(\mathbf{x})$ at $\mathbf{x}$ is then

$$\hat{z}(\mathbf{x}) = z(\mathbf{x}) + \sum_{k=1}^{K} \delta^{(k)}_{m_k(\mathbf{x})}. \tag{11}$$

5.2 Distribution contrast boosting

Here y and z are both considered to be random variables independently generated from respective distributions $p_y(y \,|\, \mathbf{x})$ and $p_z(z \,|\, \mathbf{x})$. The purpose of a contrast tree is to identify regions of $\mathbf{x}$ - space where the two distributions most differ. The goal of *distribution boosting* is to estimate a (different) transformation of z at each $\mathbf{x}$, $g_{\mathbf{x}}(z\,)$, such that the distribution of the transformed variable is the same as that of y at $\mathbf{x}$. That is,

$$p_{g_{\mathbf{x}}}(g_{\mathbf{x}}(z\,) \,|\, \mathbf{x}) = p_y(y \,|\, \mathbf{x}). \tag{12}$$

Thus, starting with z values sampled from a known distribution $p_z(z \,|\, \mathbf{x})$ at each $\mathbf{x}$, one can use the estimated transformation $\hat{g}_{\mathbf{x}}(z\,)$ to obtain an estimate $\hat{p}_y(y \,|\, \mathbf{x})$ of the y - distribution at that $\mathbf{x}$. Note that the transformation $g_{\mathbf{x}}(z\,)$ is usually a different function of z at each different $\mathbf{x}$.

The z - values within each region $R_m^{(1)}$ of a contrast tree can be transformed $z^{(1)} = g_m^{(1)}(z)$ $(\mathbf{x} \in R_m^{(1)})$ so that the discrepancy (8) with y is zero in that region. The transformation is given by

$$g_m^{(1)}(z) = \hat{F}_y^{-1}\left(\hat{F}_z(z)\right) \tag{13}$$

where $\hat{F}_y(y)$ is the empirical cumulative distribution of y for $\mathbf{x} \in R_m^{(1)}$ and $\hat{F}_z(z)$ is the corresponding distribution of z for $\mathbf{x} \in R_m^{(1)}$. This transformation function is represented by the quantile-quantile (QQ) plot of y versus z in the region.

As with estimation boosting, the distribution of the modified (transformed) variable $z^{(1)}$ can then be contrasted with that of y using another contrast tree. This produces another region set $\{R_m^{(2)}\}_1^M$ where the distributions of y and $z^{(1)}$ differ. This discrepancy (8) can be removed by transforming $z^{(1)}$ to match the distribution of y in each new region $z^{(2)} = g_m^{(2)}(z^{(1)})$ $(\mathbf{x} \in R_m^{(2)})$. These in turn can be contrasted with y producing new regions each with a corresponding transformation function. Such distribution boosting iterations can be continued K times until the discrepancy between the distributions of $z^{(K)}$ and y becomes small in each new region. As with estimation, moderating the learning rate by shrinking each estimated transformation function towards identity $g_m^{(k)}(z) \leftharpoonup (1-\alpha)\, z + \alpha\, g_m^{(k)}(z)$ usually increases accuracy at the expense of computation (more transformations).

Predicting $p_y(y \,|\, \mathbf{x})$ starts with a sample $\{z_i\}_1^n$ drawn from the specified distribution of z, $p_z(z \,|\, \mathbf{x})$, at $\mathbf{x}$. This $\mathbf{x}$ lies within one of the regions $m_k(\mathbf{x})$ of each contrast tree k with corresponding transformation function $g^{(k)}_{m_k(\mathbf{x})}(\cdot)$. A given value of z can be transformed to a estimated value for y, $\hat{y} = \hat{g}_{\mathbf{x}}(z\,)$, where

$$\hat{g}_{\mathbf{x}}(z\,) = g^{(K)}_{m_K(\mathbf{x})}(g^{(K-1)}_{m_{K-1}(\mathbf{x})}(g^{(K-2)}_{m_{K-2}(\mathbf{x})} \cdots g^{(1)}_{m_1(\mathbf{x})}(z))). \tag{14}$$

That is, the transformed output of each successive tree is further transformed by the next tree in the boosted sequence. A different transformation is chosen at each step depending on the region of the corresponding tree containing the particular joint values of the predictor variables $\mathbf{x}$. With K trees each containing M regions (terminal nodes) there are $M^{\,K}$ potentially different transformations $\hat{g}_{\mathbf{x}}(z\,)$ each corresponding to different values of $\mathbf{x}$. To the extent the overall transformation estimate $\hat{g}_{\mathbf{x}}(z\,)$ is accurate, the distribution of the transformed sample $\{\hat{y}_i = \hat{g}_{\mathbf{x}}(z_i\,)\}_1^n$ can be regarded as being similar to that of y at $\mathbf{x}$, $p_y(y \,|\, \mathbf{x})$. Statistics computed from the values of $\hat{y}$ estimating selected properties of its distribution, or the distribution itself, can be regarded as estimates of the corresponding quantities for $p_y(y \,|\, \mathbf{x})$.

6 Diagnostics

In this section we illustrate use of contrast trees as diagnostics for uncovering and understanding the lack-of-fit of predictive models for classification and conditional distribution estimation. Quantile regression models are examined in the Supporting Information. All predictive models used for illustration were applied using their respective default procedure parameter settings.

6.1 Classification

Contrast tree classification diagnostics are illustrated on the census income data obtained from the Irvine Machine Learning repository (Kohvai 1996). This data sample, taken from 1994 US census data, consists of observations from 48842 people divided into a training set of 32561 and an independent test set of 16281. The outcome variable y is binary and indicates whether or not a person's income is greater than \$50000 per year. There are 14 predictor variables $\mathbf{x}$ consisting of various demographic and financial properties associated with each person. Here we use contrast trees to diagnose the classification predictions of gradient boosted regression trees (Friedman 2001).

The predictive model produced by the gradient boosting procedure applied to the training data set produced an error rate of 13% on the test data. This quantity is the expected error as averaged over all test set predictions. It may be of interest to discover certain $\mathbf{x}$ - values for which expected error is much higher or lower. This can be ascertained by

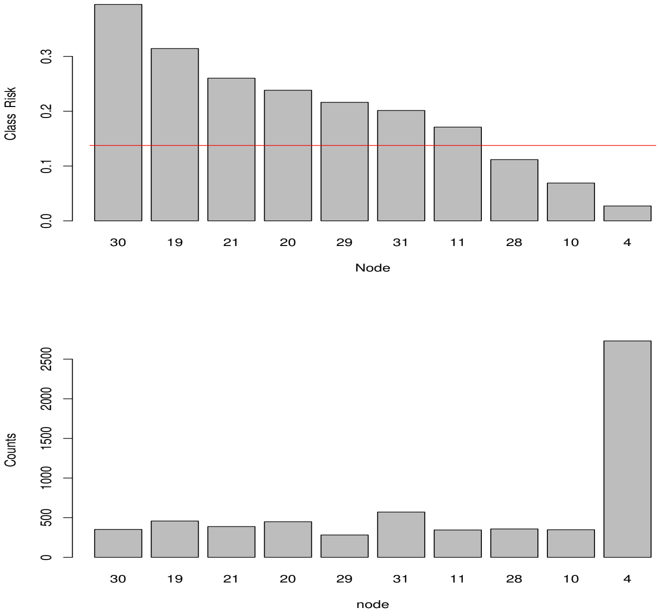

Figure 1: Misclassification risk (error rate) upper, and observation count lower, of classification contrast tree regions on census income data.

contrasting the binary outcome variable y with the model prediction z.

A natural discrepancy measure for this application is misclassification risk (error rate) in each region R_m

$$d_m = \frac{1}{N_m} \sum_{i \in R_m} I(y_i \neq z_i). \tag{15}$$

The goal in applying contrast trees is to uncover regions in $\mathbf{x}$ - space with exceptionally high values of (15). For this purpose the test data set was randomly divided into two parts of 10000 and 6281 observations respectively. A ten region contrast tree was built on the 10000 test data set. Figure 1 summarizes these regions using the separate 6281 observation data set. The upper barplot shows the misclassification risk of the gradient boosting classifier in each region ordered from largest to smallest. The lower barplot indicates the observation count in each respective region. The number below each bar is simply the contrast tree node identifier for that region. The horizontal (red) line indicates the 13% average error rate.

As Fig. 1 indicates the contrast tree has uncovered many regions with substantially higher error rates than the overall average and several others with substantially lower error rates. The lowest error region covers 43% of the test set observations with an average error rate of 2.7%. The highest error region covering 5.6% of the data has an average error rate of 41% .

Each of the regions represented in Fig. 1 are easily described. For example, the rule defining the lowest error region is

Node 4
relationship $\in$ {Own-child, Husband, Not-in-family,
Other-relative}
&
education ≤ 12

Predictions satisfying that rule suffer only a 2.7% average error rate. Predictions satisfying the rule defining the highest error region

Node 30
relationship $\notin$ {Own-child, Husband, Not-in-family,
Other-relative}
&
occupation $\in$ { Exec-managerial, Transport-moving,
Armed-Forces }
&
education ≤ 12

have a 41% average error rate. Thus confidence in salary predictions for people in node 4 might be higher than for those in node 30.

6.2 Probability estimation

The discrepancy measure (15) is appropriate for procedures that predict a class identity and the corresponding contrast tree attempts to identify $\mathbf{x}$ - values associated with high levels of misclassification. Some procedures such as gradient boosting return estimated class probabilities at each $\mathbf{x}$ which are then thresholded to predict class identities. In this case the probability estimate contains information concerning expected classification accuracy. The closer the respective class probabilities are to each other the higher is the likelihood of misclassification. This shifts the issue from classification accuracy to probability estimation accuracy which can be assessed with a contrast tree.

For binary classification a natural discrepancy for probability estimation is (6) where $y \in \{0, 1\}$ is the binary outcome variable and $0 \leq z \leq 1$ is its predicted probability $\widehat{\Pr}(y = 1)$. This measures the difference between the empirical probability of $y = 1$ in region R_m and the corresponding average probability prediction z in that region. The gradient boosting probability estimates were based on the training data set. A ten terminal node contrast tree was built on the census income data using the 10000 observation test data set with corresponding node statistics evaluated on the separate 6281 observation test data set.

The top frame of Fig. 2 shows the empirical probability $y = 1$ (blue) and the average gradient boosting prediction z (red) within each region of the resulting contrast tree.

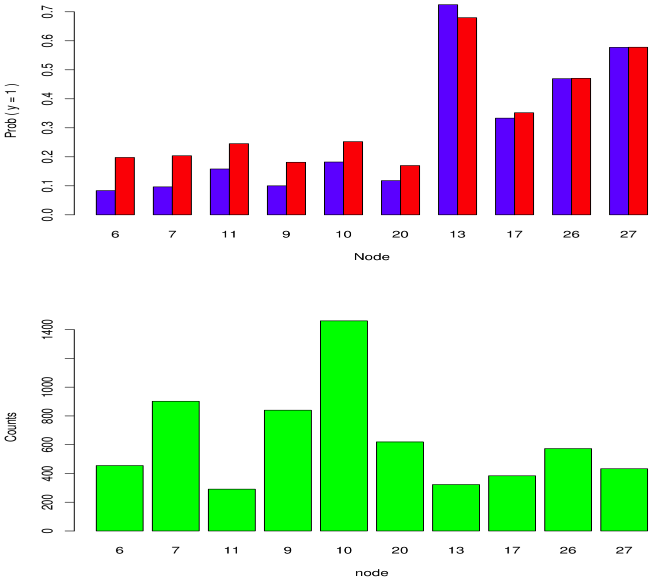

Figure 2: Census income data. Upper frame: fraction of positive observations (blue) and mean probability prediction (red) for probability contrast tree regions. Lower frame: observation count in each region.

The bottom frame shows the number of counts in each corresponding region. One sees a general trend of over-smoothing. The largest probability is being under-estimated whereas the smaller ones are substantially over-estimated by the gradient boosting procedure. As above each of these regions is defined by simple rules based on the values of a few predictor variables.

A convenient way to summarize the overall results of a contrast tree is through its corresponding lack-of-fit contrast curve. For each region R_m containing N_m counts, the observation weighted average of its discrepancy d_m and those with higher discrepancy

$$\bar{d}_m = \sum_{d_j \geq d_m} d_j N_j / \sum_{d_j \geq d_m} N_j \tag{16}$$

is plotted on the vertical axis. The fraction of observations in those same regions

$$f_m = \frac{1}{N} \sum_{d_j \geq d_m} N_j \tag{17}$$

is plotted along the horizontal axis. The left most point on each curve thus represents the discrepancy value of the largest discrepancy region of its corresponding tree. The right most point gives the discrepancy averaged over all regions. Intermediate points give average discrepancy over the highest discrepancy regions containing the corresponding fraction of observations.

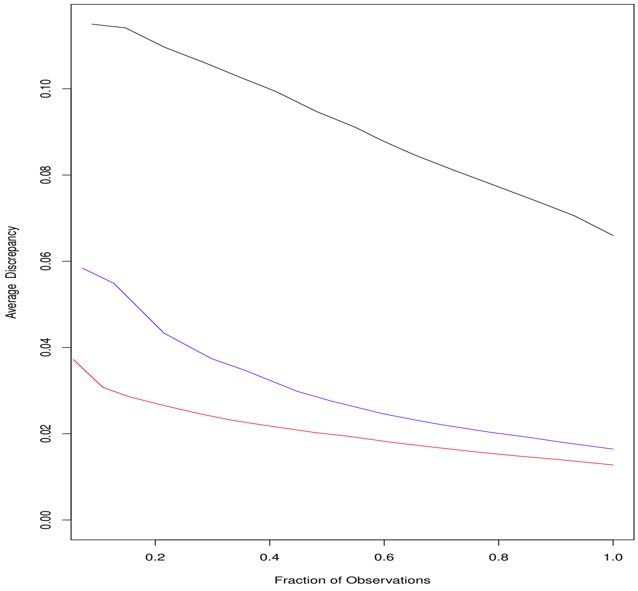

Figure 3: Census income data. Lack-of-fit contrast curves comparing accuracy of $\Pr(y = 1)$ estimates by logistic gradient boosting (black), random forests (blue), and probability gradient boosting (red).

The black curve in Fig. 3 shows the lack-of-fit contrast curve for the gradient boosting estimates based on a 50 node contrast tree built in the same manner as the one shown in Fig. 2. Its error in estimated probability averaged over all test set predictions is seen to be 0.066 (extreme right). The error corresponding to the largest discrepancy region (extreme left) is 0.115. The blue curve is the corresponding lack-of-fit contrast curve for random forest probability prediction (Breiman 2001). Its average error is less than one third of that for gradient boosting and its worst error is 50% less.

The contrast tree as represented in Fig. 2 suggests that the problem with the gradient boosting procedure here is oversmoothng. It is failing to accurately estimate the extreme probability values. Gradient boosting for binary probability estimation generally uses a negative Bernoulli log–likelihood loss function based on a logistic distribution. The logistic transformation to modeling on the log-odds scale inhibits the estimation of extreme probability values. Random forests use regression trees that model directly on the probability scale using squared–error loss. This suggests that using a similar approach with gradient boosting for this problem may improve performance, especially at the extreme values.

The red curve in Fig. 3 shows the corresponding lack-of-fit contrast curve for direct probability estimation with gradient boosting using squared–error loss. This change has dramatically improved accuracy of gradient boosting probability estimates. Both its average and maximum discrepancies are

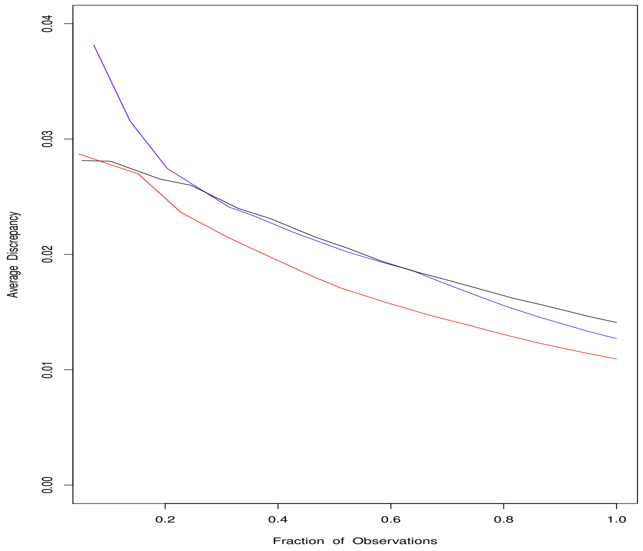

Figure 4: Census income data. Lack-of-fit contrast curves comparing accuracy of $\Pr(y = 1)$ estimates after applying contrast boosting to the output of logistic gradient boosting (black), random forests (blue), and probability gradient boosting (red).

seen to be at least four times smaller than those using the approach based on logistic regression.

Figure 4 shows the corresponding test data results of applying contrast *boosting* (Section 5.1) to the training data output of each of the methods shown in Fig. 3. Comparing the two figures one sees that the accuracy of logistic gradient boosting is dramatically improved while that of random forest is substantially improved. The improvement to probability gradient boosting using squared-error loss is seen to be moderate.

Table 1

Classification error rates corresponding to several probability estimation methods.

Method	Error rate
Logistic gradient Boosting	13.0%
Probability gradient Boosting	12.9%
Random Forest	13.6%
Prob. grad. Boost + Contrast	12.8%

Table 1 shows classification error rate for each of the three original methods plus that of the best contrast boosting result. They are all seen to be very similar. This illustrates that prediction error on the random outcome variable can be a very poor proxy for estimation accuracy of the distribution mean ($\Pr(y = 1)$). Here the over–smoothing of probability estimates caused by modeling log-odds does not change many class assignments. In some applications accurate estimation of extreme probabilities is important, such as with highly asymmetric misclassification losses. In such cases directly estimating on the probability scale may be superior to indirectly estimating on the log-odds scale.

6.3 Conditional distributions

Here we consider the case in which both y and z are considered to be random variables independently drawn from respective distributions $p_y(y \,|\, \mathbf{x})$ and $p_z(z \,|\, \mathbf{x})$. Interest is in contrasting these two distributions as functions of $\mathbf{x}$. Specifically we wish to uncover regions of $\mathbf{x}$ - space where the distributions most differ. For this we use contrast trees (Section 3) with discrepancy measure (8).

A well known way to approximate $p_y(y \,|\, \mathbf{x})$ under the assumption of homoskedasticity is through the residual bootstrap (Efron and Tibshirani 1994). One obtains a location estimate such as the conditional median $\hat{m}(y \,|\, \mathbf{x})$ and forms the data residuals $r_i = y_i - \hat{m}(y \,|\, \mathbf{x}_i)$ for each observation $1 \leq i \leq N$. Under the assumption that the conditional distribution of r, $p_r(r \,|\, \mathbf{x})$, is independent of $\mathbf{x}$ (homoskedasticity) one can draw random samples from $p_y(y \,|\, \mathbf{x}_i)$ as $y_i = \hat{m}(y \,|\, \mathbf{x}_i) + r_{\pi(i)}$ where $\pi(i)$ is random permutation of the integers $i \in [1, N]$. These samples can then be used to derive various regression statistics of interest.

A fundamental ingredient for the validity of residual bootstrap approach is the homoskedasticity assumption. Here we test this on the online news popularity data set (Fernandes, Vinagre and Cortez, 2015) also available from the Irvine Machine Learning Data Repository. It summarizes a heterogeneous set of features about articles published by Mashable web site over a period of two years. The goal is to predict the number of shares y in social networks (popularity). There are $N = 39797$ observations (articles). Associated with each are $p = 59$ attributes to be used as predictor variables $\mathbf{x}$. These are described at the download web site. Gradient boosting was used to estimate the median function $\hat{m}(y \,|\, \mathbf{x})$, and $\{z_i\}_{i=1}^N$ was taken as a corresponding residual bootstrap sample to be contrasted with y.

Figure 5 shows quantle-quantile (QQ)-plots of y versus z for the nine highest discrepancy regions of a 50 node contrast tree. The red line represents equality. One sees that there are $\mathbf{x}$ - values (regions) where the distribution of y is very different from its residual bootstrap approximation z; homoskedasticity is rather strongly violated. The average discrepancy (8) over all 50 regions is 0.19.

The outcome variable y (number of shares) is strictly positive and its marginal distribution is highly skewed toward larger values. In such situations it is common to model its logarithm. Figure 6 shows the corresponding results for contrasting the distribution of $\log_{10}(y)$ with its residual bootstrap counterpart. Homoskedasticity appears to more closely hold on the logarithm scale but there are still regions of $\mathbf{x}$ - space

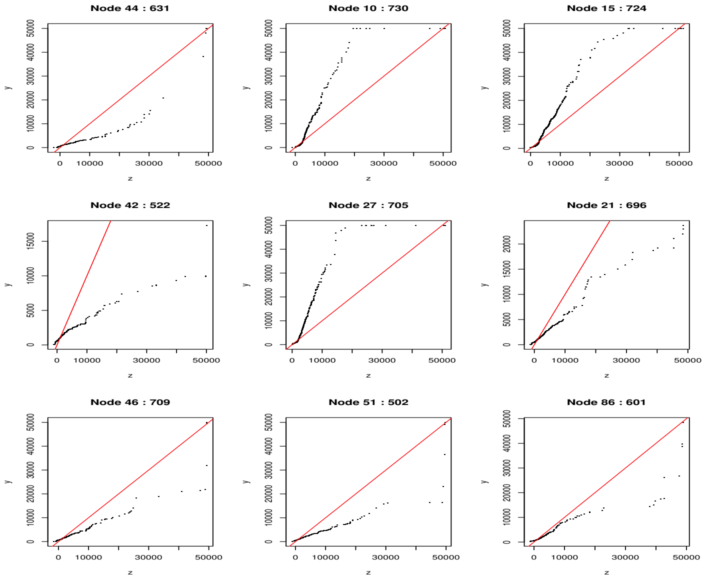

Figure 5: QQ–plots of y versus parametric bootstrap z distributions for the nine highest discrepancy regions of a 50 node contrast tree using online news popularity data. The red line represents equality.

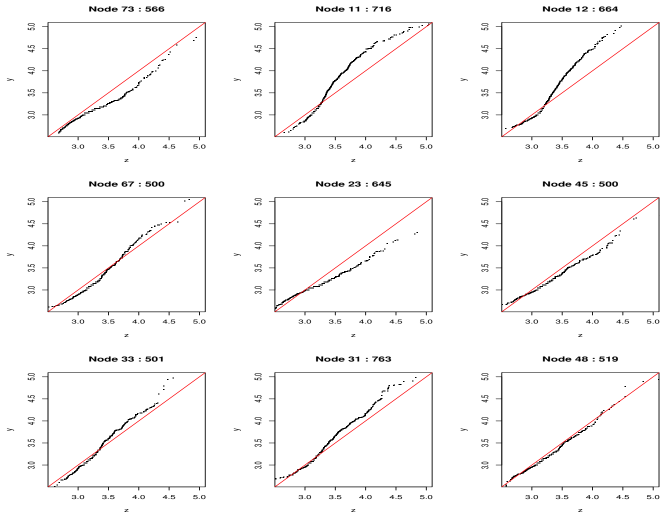

Figure 6: QQ–plots of $\log_{10}(y)$ versus corresponding parametric bootstrap z distributions for the nine highest discrepancy regions of a 50 node contrast tree using online news popularity data. The red line represents equality.

where the approximation is not good. Here the average discrepancy (8) over all 50 regions is 0.13. A null distribution for average discrepancy under the hypothesis of homoskedasticity can be obtained by repeatedly contrasting pairs of randomly generated $\log_{10}(y)$ residual bootstrap distributions. Based on 50 replications, this distribution had a mean of 0.078 with a standard deviation of 0.003.

7 Distribution boosting – simulated data

The notion of distribution boosting (Section 5.2) is sufficiently unusual that we first illustrate it on simulated data where the estimates $\hat{p}_y(y \,|\, \mathbf{x})$ can be compared to the true data generating distributions $p_y(y \,|\, \mathbf{x})$. Distribution boosting applied to the online news popularity data described in Section 6.3 is presented in the Supporting Information.

7.1 Data

There are $N = 25000$ training observations each with a set of $p = 10$ predictor variables $\mathbf{x}_i$ randomly generated from a standard normal distribution. The outcome variable $y \,|\, \mathbf{x}$ is generated from a transformed *asymmetric* logistic distribution (Friedman 2018)

$$y = h(f(\mathbf{x}) + \eta(\mathbf{x})) \tag{18}$$

with the random component being $\eta(\mathbf{x}) = -|\,\varepsilon\,| \cdot s_l(\mathbf{x})$ with probability $P_l = s_l(\mathbf{x})/(s_l(\mathbf{x}) + s_u(\mathbf{x}))$ and $\eta(\mathbf{x}) = +|\,\varepsilon\,| \cdot s_u(\mathbf{x})$ with probability $s_u(\mathbf{x})/(s_l(\mathbf{x}) + s_u(\mathbf{x}))$. Here ε is a standard logistic random variable. The transformation $h(z)$ is taken to be

$$h(z) = sign(z)\,(0.5\,|\,z\,| + 1.5\; z^2). \tag{19}$$

The untransformed mode $f(\mathbf{x})$ and lower/upper scales $s_l(\mathbf{x}) \,/\, s_u(\mathbf{x})$ are each different functions of the ten predictor variables $\mathbf{x}$. The simulated mode function is taken to be

$$f(\mathbf{x}) = \sum_{j=1}^{10} c_j\, B_j(x_j) \,/\, std_{x_j}(B_j(x_j)\,) \tag{20}$$

with the value of each coefficient c_j being randomly drawn from a standard normal distribution. Each basis function takes the form

$$B_j(x_j) = sign(x_j)\,|\,x_j\,|^{r_j} \tag{21}$$

with each exponent r_j being separately drawn from a uniform distribution $r_j \sim U(0,2)$. The denominator in each term of (20) prevents the suppression of the influence of highly nonlinear terms in defining $f(\mathbf{x})$.

The scale functions are taken to be $s_l(\mathbf{x}) = 0.2 + \exp\,(t_l(\mathbf{x}))$ and $s_u(\mathbf{x}) = 0.2 + \exp(t_u(\mathbf{x}))$ where the log–scale functions

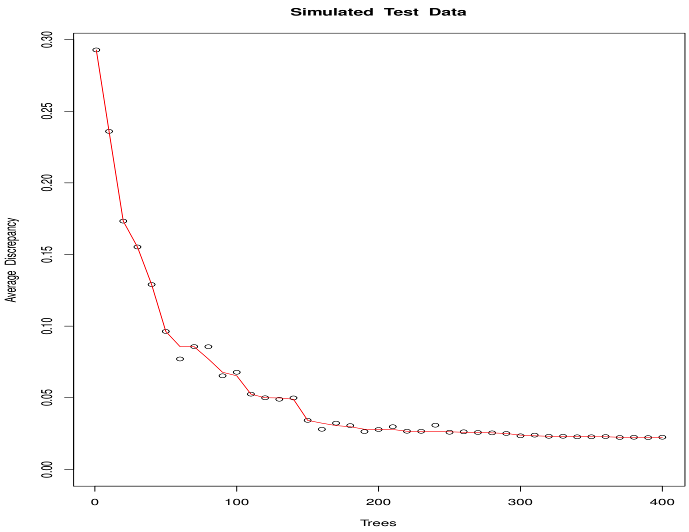

Figure 7: Test data discrepancy averaged over the terminal nodes (regions) of successive contrast trees for the first and then every tenth iteration for 400 iterations of distribution boosting on simulated training data. The solid red curve is a running median smooth.

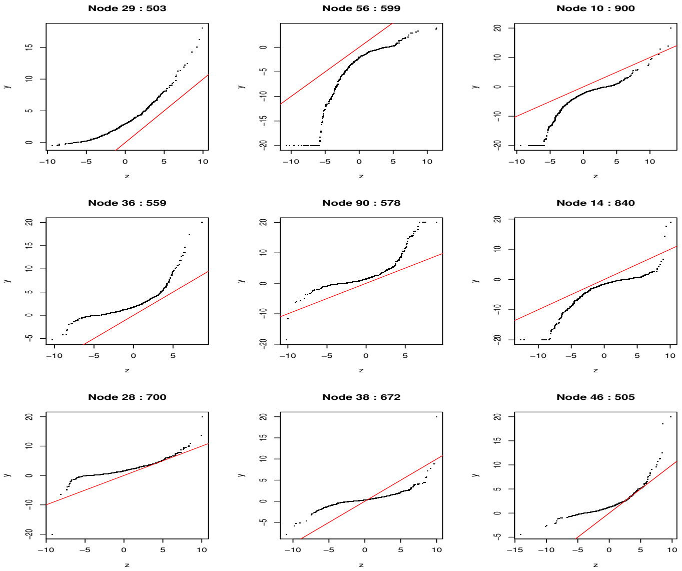

Figure 8: QQ–plots of y versus z (normal) for the nine highest discrepancy regions of a 10 node contrast tree on the simulated test data set. The red lines represent equality.

$t_l(\mathbf{x})$ and $t_u(\mathbf{x})$ are constructed in the same manner as (20) (21) but with different randomly drawn values for the 20 parameters $\{c_j, r_j\}_1^{10}$ producing different functions of $\mathbf{x}$. The average pair-wise absolute correlation between the three functions is 0.18. The overall resulting distribution $p(y \,|\, \mathbf{x})$ (18–21) has location, scale, asymmetry, and shape being highly dependent on the joint values of the predictors $\mathbf{x}$ in a complex and unrelated way.

7.2 Conditional distribution estimation

Distribution boosting is applied to this simulated data to estimate its distribution $p_y(y \,|\, \mathbf{x})$ as a function of $\mathbf{x}$. For each observation the contrasting random variable z is taken to be independently generated from the same normal distribution, $z \,|\, \mathbf{x} \sim N(\bar{y}, \sigma_y^2)$, independent of $\mathbf{x}$. Here $\bar{y}$ and σ_y^2 are the mean and variance of the marginal y–distribution. The goal is to produce an estimated transformation of z, $\hat{y} = \hat{g}_{\mathbf{x}}(z)$, at each $\mathbf{x}$ such that $p_{\hat{y}}(\hat{y} \,|\, \mathbf{x}) = p_y(y \,|\, \mathbf{x})$. To the extent the estimate $\hat{g}_{\mathbf{x}}(z)$ accurately reflects the true transformation function $g_{\mathbf{x}}(z)$ at each $\mathbf{x}$ one can apply it to a sample drawn from $z \sim N(\bar{y}, \sigma_y^2)$ to produce a corresponding sample drawn from the distribution $y \sim p_y(y \,|\, \mathbf{x})$. This sample can then be used to plot that distribution or compute the value of any of its properties.

Figure 7 plots the average terminal node discrepancy (8) for 400 iterations of distribution boosting applied to the training data, as evaluated on a 25000 observation independent "test" data set generated from the same joint $(\mathbf{x}, y)$ - distribution (18–21). Results are shown for the first and then every tenth successive tree. The red line is a running median smooth. The test set discrepancy is seen to generally decrease with increasing number of trees. There is a diminishing return after about 200 iterations (trees).

Note that with contrast boosting average tree discrepancy on test or even training data does not necessarily decrease monotonically with successive iterations (trees). Each contrast tree represents a greedy solution to a non convex optimization with multiple local optima. As a consequence the inclusion of an additional tree can, and often does, increase average discrepancy of the current ensemble. Boosting is continued as long as there is a general downward trend in average tree discrepancy.

Lack-of-fit to the data of any model for the distribution $p_y(y \,|\, \mathbf{x})$ can be assessed by contrasting y with a sample drawn from that distribution. Figure 8 shows QQ–plots of y versus initial z (everywhere the same normal) for the nine highest discrepancy regions of a 10 node tree contrasting the two quantities on the test data set. The red lines represent equality. One sees that $p_y(y \,|\, \mathbf{x})$ is here far from being everywhere the same normal.

For the distribution boosted model $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ lack-of-fit can be assessed by contrasting the distributions of y and $\hat{y}$ with a contrast tree using the test data set. Figure 9 shows QQ–plots of y versus $\hat{y}$ for the nine highest discrepancy regions of a 10 node tree contrasting the two quantities on the test data set. The red lines represent equality. The transformation $\hat{g}_{\mathbf{x}}(z)$ at each separate $\mathbf{x}$ - value was evaluated using the 400 tree

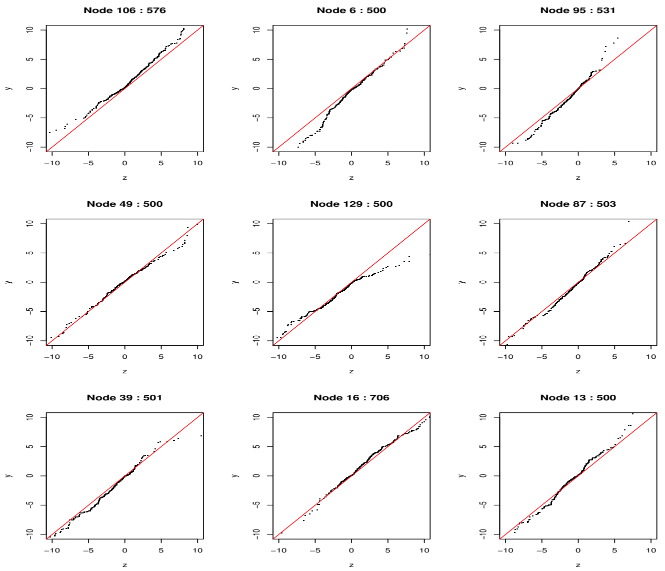

Figure 9: QQ–plots of y versus $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ for the nine highest discrepancy regions of a 10 node contrast tree on the simulated test data set. The red lines represent equality.

model built on the training data. The nine highest discrepancy regions shown in Fig. 9 together cover 27% of the data. They show that while the transformation model fits most of the test data quite well, it is not everywhere perfect. There are minor departures between the two distributions in some small regions. However these discrepancies appear in sparse tails where QQ–plots themselves can be unstable.

A measure of the difference between the estimated and true CDFs at each $\mathbf{x}$ can be defined as

$$Diff\,(\mathbf{x}) = \sqrt{\frac{1}{100}\sum_{j=1}^{100}(\,CDF_{\mathbf{x}}(u_j) - \widehat{CDF}_{\mathbf{x}}(u_j)\,)^2} \qquad (22)$$

where $CDF_{\mathbf{x}}$ is the true cumulative distribution of $y \,|\, \mathbf{x}$ computed from (18–21) and $\widehat{CDF}_{\mathbf{x}}$ is the corresponding estimate from the distribution boosting model. The 100 evaluation points $\{u_j\}_1^{100}$ are a uniform grid between the 0.001 and 0.999 quantiles of the true distribution $CDF_{\mathbf{x}}$.

Figure 10 summarizes the overall accuracy of the distribution boosting model. The upper left frame shows a histogram of the distribution of (22) for observations in the test data set. The 50, 75 and 90 percentiles of this distribution are respectively 0.0352, 0.0489 and 0.0773 indicated by the red marks. The remaining plots show estimated (black) and true (red) distributions for the three observations with (22) equal to these respective percentiles. Thus 50% of the estimated distributions are closer to the truth than that shown in the upper right frame. Seventy five percent are closer than that shown in the lower left frame, and 90% are closer than that seen in the lower right frame.

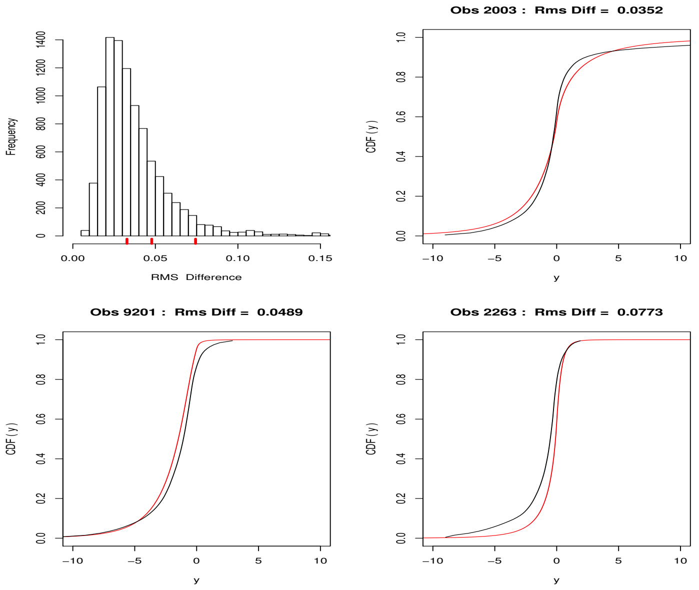

Figure 10: Upper left: CDF error (22) distribution for simulated data. Upper right: estimated (black) and true (red) CDFs for observation with median error. Lower: corresponding plots for 75% and 90% decile errors.

Distribution boosting produces an estimate for the full distribution of $y \,|\, \mathbf{x}$ by providing a function $\hat{g}_{\mathbf{x}}(z)$ that transforms a random variable z with a known distribution $p_z(z \,|\, \mathbf{x})$ to the estimated distribution $\hat{p}_y(y \,|\, \mathbf{x})$. One can then easily compute any statistic $\hat{S}(\mathbf{x}) = S[\,\hat{p}_y(y \,|\, \mathbf{x})]$, which can be used as an estimate for the value of the corresponding quantity $S(\mathbf{x}) = S[$ $p_y(y \,|\, \mathbf{x})]$ on the actual distribution. For some quantities $S(\mathbf{x})$, an alternative is to directly estimate them by minimizing empirical prediction risk based on an appropriate loss function

$$\hat{S}(\mathbf{x}) = \arg\min_{f \in \Im} \frac{1}{N}\sum_{i=1}^{N} L(y_i, f(\mathbf{x}_i)) \qquad (23)$$

where $\Im$ is the function class associated with the learning method. Here we compare distribution boosting (DB) estimates of the quartiles $Q_p(\mathbf{x})$, $p \in [0.25, 0.5, 0.75]$, with those of gradient boosting quantile regression (GB), which uses loss

$$L_p(y, z) = (1-p)\,(z-y)_+ + p\,(y-z)_+, \qquad (24)$$

on the simulated data set where the truth is known.

Figure 11 shows true versus predicted values for each of the two methods (rows) on the three quartiles (columns). The red lines represent a running median smooth and the blue lines show equality. The average absolute error AAE associated with each of these plots is

$$AAE(h, v) = mean(|\,h - v\,|)/mean(|\,v - median(v)\,|) \quad (25)$$

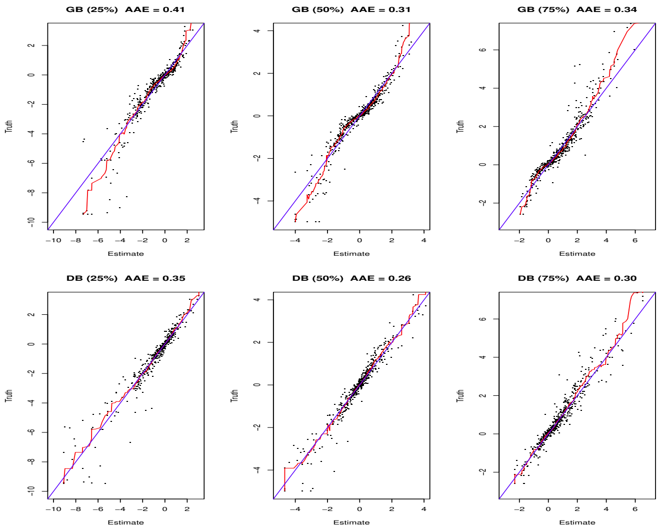

Figure 11: Predicted versus true values for the three quartiles as functions of $\mathbf{x}$ (columns) for gradient boosting quantile regression (upper row) and distribution boosting (lower row) on the simulated data. The red lines represent a running median smooth and the blue lines show equality.

where h is the quantity plotted on the horizontal and v the vertical axes. The quantile values derived from the estimates of the full distribution (bottom row) are here seen to be somewhat more accurate than those obtained from gradient boosting quantile regression (top row).

With quantile regression each quantile is estimated separately without regard to estimates of other quantiles. Distribution boosting quantile estimates are all derived from a common probability distribution and thus have order constraints imposed among them. For example, two quantile estimates have the property $\hat{Q}_p(\mathbf{x}) < \hat{Q}_{p'}(\mathbf{x})$ for all $p < p'$ at any $\mathbf{x}$. These implicit constraints can improve accuracy especially when the quantile estimates are being used to compute quantities derived from them.

There is an additional advantage of computing quantities such as means or quantiles from the estimated conditional distributions $\hat{S}(\mathbf{x}) = S[\,\hat{p}_y(y \,|\, \mathbf{x})]$. As noted in Section 4, distribution contrast trees can be constructed in the presence of arbitrary censoring or truncation. This extends to contrast boosted distribution estimates $\hat{p}_y(y \,|\, \mathbf{x})$ and any quantities derived from them. This in turn allows application to ordinal regression which can be considered a special case of interval censoring (Friedman 2018).

8 Discussion

When a discrepancy measure takes the special form of an average over individual observation losses, such as (5) or model residuals (15), one can use an ordinary regression tree (or other standard learning methods) to directly model the discrepancy as a function of $\mathbf{x}$. This may uncover $\mathbf{x}$ - values corresponding to relatively high discrepancy. However, such a strategy is not focused on this task but rather on trying to approximate discrepancy over entire distribution of $\mathbf{x}$ - values. The contrast tree splitting strategy (4) directly seeks high discrepancy regions regardless of local data density thereby largely ignoring the $\mathbf{x}$ - distribution. Besides increased sensitivity to high discrepancy, this property has the additional effect of rendering contrast tree based methods more robust against distribution drift. Standard leaning methods are not applicable to discrepancy measures that are *not* simple averages of single observation loss criterion, such as (6) (7) (8).

The fitting paradigm of contrast trees is somewhat different than that of ordinary machine learning. The goal of the latter is data fitting. That is to capture as much structure as possible in the relation between y and $\mathbf{x}$. The more structure captured the better the model, subject to over-fitting considerations. Over-fitting occurs when the model captures non generalizable data specific relationships. Contrast trees attempt to uncover *lack-of-fit*. The more structure they capture, the *worse* the model fits the data.

This reversal of emphasis has consequences for interpretation. With regular machine learning evaluating the quality of a model on its own training data generally produces an over optimistic measure of model quality. With contrast trees this gives a conservative overly pessimistic assessment of model accuracy, especially for large trees built with small samples. For small trees and/or large samples the effect is usually small. Using different data to construct the tree and evaluate its node statistics eliminates this bias at the cost of increased variance.

9 Related work

Regression trees have a long history in Statistics and Machine Learning. Since their first introduction (Morgan and Songquist 1963) many proposed modifications have been introduced to increase accuracy and extend applicability. See Loh (2014) for a nice survey. More recent extensions include Mediboost (Valdes *et al* 2016) and the Additive Tree (Luna *et al* 2019). All of these proposals are focused towards estimating the properties of a single outcome variable. There has been work on using trees for simultaneous estimation of several outcome variables (Segal and Xiao 2011) but there seems to have been little to no work related to applications involving contrasting two such variables.

Although not directly involving trees, Friedman and Fisher (1999) proposed using recursive partitioning strategies to identify interpretable regions in $\mathbf{x}$ - space within which the mean of a single outcome y was relatively large ("hot spots"). With a similar goal Buja and Lee (2001) proposed using ordinary regression trees with a splitting criterion based on the maximum of the two daughter node means.

Classification tree boosting was proposed by Freund and Schapire (1997). Extension to regression trees was developed by Friedman (2001). Since then there has been considerable research attempting to improve accuracy and extend its scope. See Mayr *et al* (2014) for a good summary.

Although boosted contrast trees have not been previously proposed they are generally appropriate for the same types of applications as gradient boosted regression trees, such as classification, regression, and quantile regression. They can be beneficial in applications where a contrast tree indicates lack-of-fit of a model produced by some estimation method. In such situations applying contrast boosting to the model predictions often provides improvement in accuracy.

Tree ensembles have also been applied to nonparametric conditional distribution estimation. Meinshausen (2006) used classical random forests to define local neighborhoods in $\mathbf{x}$ -space. The empirical conditional distribution of y in each such defined local region around a prediction point $\mathbf{x}$ is taken as the corresponding conditional distribution estimate at $\mathbf{x}$. Athey, Tibshirani and Wagner (2019) noted that since the regression trees used by random forests are designed to detect only mean differences the resulting neighborhoods will fail to adequately capture distributions for which higher moments are not generally functions of the mean. They proposed modified tree building strategies based on gradient boosting ideas to customize random forest tree construction for specific applications including quantile regression.

Boosted regression trees have been used as components in procedures for parametric fitting of conditional distributions and transformations. A parametric form for the conditional distribution or transformation is hypothesized and the parameters as functions of $\mathbf{x}$ are estimated by regression tree gradient boosting using negative log–likelihood as the prediction risk. See for example Mayr et al (2012), Friedman (2018), Pratola *et al* (2019), Hothorn (2019) and Mukhopadlhyay & Wang (2019). Some differences between these previous methods and the corresponding approaches proposed here include use of contrast rather than regression trees, and no parametric assumptions.

The principal benefit of the contrast tree based procedures is a lack-of-fit measure. As seen in Table 1 of Section 6.2, and in the Supporting Information, values of negative log–likelihoods or prediction risk need not reflect actual lack-of-fit to the data. The values of their minima can depend upon other unmeasured quantities. The goal of contrast trees as illustrated in this paper is to provide such a measure. Contrast trees can be applied to assess lack-of-fit of estimates produced by any method, including those mentioned above. If discrepancies are detected, contrast boosting can be employed to remedy them and thereby improve accuracy.

10 Summary

Contrast trees as described in Sections 3 and 4 are designed to provide interpretable goodness-of-fit diagnostics for estimates of the parameters of $p_y(y \,|\, \mathbf{x})$, or the full distribution. Examples involving classification, probability estimation and conditional distribution estimation were presented in Section 6. A quantile regression example is presented in the Supporting Information. Two–sample contrast trees for detecting discrepancies between separate data sets are also described in the Supporting Information.

Boosting of contrast trees is a natural extension. Given an initial estimate $\hat{z}(\mathbf{x})$ from any learning method a contrast tree can assess its goodness or lack-of-fit to the data. If found lacking, the boosting strategy attempts to improve the fit by successively modifying $\hat{z}(\mathbf{x})$ to bring it closer to the data. As seen in Fig. 3 this strategy can substantially improve prediction accuracy for some methods. The Supporting Information provides such an example involving quantile regression.

Contrast boosting the full conditional distribution is illustrated on simulated data in Section 7.2 and on actual data in the Supporting Information. Note that the conditional distribution procedure of Section 5.2 can be applied in the presence of arbitrarily censored or truncated data by employing Turnbull's (1976) algorithm to compute CDFs and corresponding quantiles.

Contrast trees and boosting inherit all of the data analytic advantages of classification and regression trees. These include handling categorical variables and missing values, invariance to monotone transformations of the predictor variables, resistance to irrelevant predictors, variable importance measures, and few tuning parameters.

Important discussions with Trevor Hastie and Rob Tibshirani on the subject of this work are gratefully acknowledged. An *R* procedure implementing the methods described herein is available.

References

[1] Anderson, T. and Darling, D. (1952). Asymptotic theory of certain "goodness-of-fit" criteria based on stochastic processes. *Ann. Sta*t. **23**, 193–212.

[2] Athey, S., Tibshirani, J., Wagner, S. (2019). Generalized random forests. *Ann. Stat.* **47**(2), 1148-1178.

[3] Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees.* Chapman and Hall.

[4] Breiman, L. (2001). Random forests. Machine Learning **45**, 5-32.

[5] Buja, A and Lee, Y. (2001). Data mining criteria for tree-based regression and classification. *Proceedings of KDD 2001*, 27–36.

[6] Efron, B. (1979). Bootstrap Methods: Another look at the jackknife. Ann. Stat. **7**, 1-26.

[7] Efron, B. and Tibshirani, R. (1994). *An Introduction to the Bootstrap.* Springer.

[8] Fernandes, K., Vinagre, P. and Cortez, P. (2015). A proactive intelligent decision support system for predicting the popularity of online news. *Proceedings of the 17th EPIA 2015 - Portuguese Conference on Artificial Intelligence.* September, Coimbra, Portugal.

[9] Freund, Y. and Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *J. Computer and System Sciences* **55**, 119-139.

[10] Friedman, J. and Fisher, N. (1999) Bump hunting in high-dimensional data. *Statistics and Computing*, **9**, 123-143.

[11] Friedman, J. (2001). Greedy function approximation: a gradient boosting machine. *Ann. Stat.* **29** 1189-1232.

[12] Friedman, J. (2018). Predicting regression probability distributions with imperfect data through optimal transformations. Stanford University Statistics Technical Report arXiv: 2001, 10102 [stat. ML]

[13] Hothorn, T. (2019). Transformation boosting machines. *Statistics and Computing.* https://doi.org/10.1007/s11222-019-09870-4.

[14] Kohavi, R. (1996). Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 1996

[15] Loh, W. (2014). Fifty years of classification and regression trees. *Inter. Statist. Rev.* **82**, 3, 329–348.

[16] Luna, J., Gennatas, E., Eaton, E., Diffenderfer, E., Ungar, L., Jensen, S., Simone, C., Friedman, J., Valdes, G. (2019). The additive tree. *Proc. Nat. Acad. Sci.***116** (40), 19887-19893.

[17] Mayr, A., Fenske, N., Hofner, B., Kneib, T., Schmid, M. (2012). GAMLSS for high dimensional data–a flexible approach based on boosting. *J. R. Stat. Soc. Ser. C* (Appl. Stat) **61**(3), 403–427.

[18] Mayr, A., Binder, H., Gefeller, O., and Schmid, M. (2014). The evolution of boosting algorithms from machine learning to statistical modelling. *Methods Inf. Med.* **53**, 419–427.

[19] Meinshausen, M. (2006). Quantile random forests. *J. Machine Learning Research* **7**, 983–999.

[20] Morgan, J. and Sonquist, J. (1963). Problems in the analysis of survey data, and a proposal. *J. Amer. Statist. Assoc.* **58**, 415–434.

[21] Mukhopadhyay, S. and Wang, K. (2019). On the problem of relevance in statistical inference. *J. Amer. Statist. Assoc.*(submitted).

[22] Pratola, M. T., Chipman, H. A, George, E. I., and McCulloch, R. E. (2019). Heteroscedastic BART via multiplicative regression trees, *J. of Comput. and Graphical Statist.*, DOI: 10.1080/10618600.2019.1677243

[23] Segal, M. and Xiao, Y. (2011). Multivariate Random Forests. *WIREs Data Mining and Knowl. Discov.* **1**, 80–87.

[24] Turnbull, B. W. (1976). The empirical distribution function with arbitrarily grouped, censored and truncated data. J. Royal *Statist. Soc.* B **38**, 290-295.

[25] Valdes, G., Luna, J., Eaton, E., Simone, C , Ungar, L. and Solberg, T. (2016). MediBoost: a patient stratification tool for interpretable decision making in the era of precision medicine. *Scientific Reports* **6**, Article number: 37854.

Supporting Information
Contrast Trees and Distribution Boosting

Jerome H. Friedman

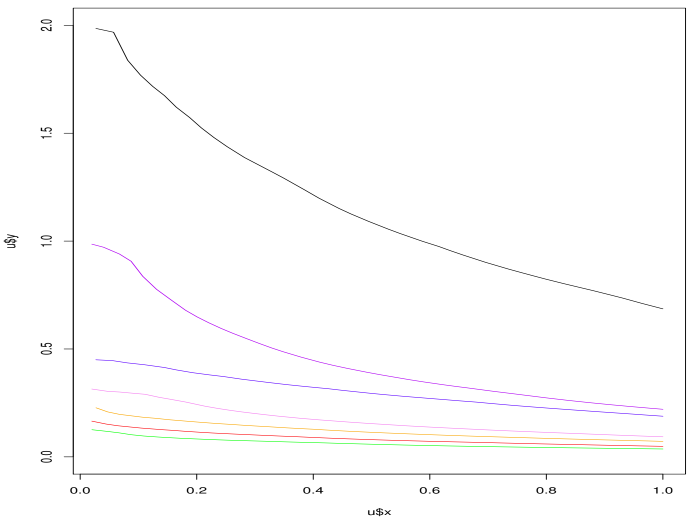

Figure S1: Lack-of-fit contrast curves on simulated data. Black: constant fit, purple: single CART tree, blue: linear model, violet: random forest, orange: squared-error and red: absolute loss gradient boosting, green: truth.

S1 Lack-of-fit estimation

Here contrast tree lack-of-fit estimates are compared with known truth on simulated data. There are $N = 25000$ observations each with $p = 10$ predictor variables $\mathbf{x}$ randomly generated from a standard normal distribution. The outcome y is generated from a simple model

$$y = f(\mathbf{x}) + s(\mathbf{x}) \cdot \varepsilon$$

with ε a standard normal random variable. The location $f(\mathbf{x})$ and log-scale $\log(s(\mathbf{x}))$ functions are given by (20) (21) with different randomly generated parameters. The correlation between the two functions over the data is $cor(f(\mathbf{x}), s(\mathbf{x})) = 0.06$. The signal/noise is $IQR(f(\mathbf{x}))/(2 \cdot med(s(\mathbf{x}))) = 3$. The goal is to estimate the location function $f(\mathbf{x})$.

Lack-of-fit contrast curves (16) (17) for six methods are shown in Fig. S1. The methods are (top to bottom): black constant fit (global mean), purple single CART tree, blue linear least-squares fit, violet random forest, orange squared-error and red absolute loss gradient boosting. The bottom green curve represents the lack-of-fit contrast curve for the true mean function $f(\mathbf{x})$ on these data. All curves were evaluated on a separate 25000 observation test data set not used to train the respective models.

Table S1
RMS estimation error and contrast tree RMS discrepancy for several methods

Method	RMS Error	Discrepancy
constant	0.99	0.86
CART tree	0.57	0.34
linear model	0.33	0.23
random forest	0.21	0.13
sqr-error boost	0.15	0.090
abs-error boost	0.11	0.063
truth	0	0.046

Since the data are simulated and truth $f(\mathbf{x})$ is here known one can directly compute root-mean-squared estimation error

$$RMSE = \sqrt{mean((f(\mathbf{x}) - \hat{f}(\mathbf{x}))^2)}$$

for each method. This is shown in Table S1 (second column) for each method (first column). The third column shows the root-mean-squared discrepancy over the same test observations calculated from the respective contrast trees for each method. The discrepancy associated with an observation is that of the contrast tree region that contains it.

Except for the (usually unknown) true mean function $f(\mathbf{x})$ itself, empirical contrast tree discrepancy is generally smaller than RMS error. This is because a finite region contrast tree cannot capture actual discrepancy in perfect detail. Failure to capture this structure results in under estimation of discrepancy for all methods (see Section 8). Here discrepancy as computed on the data and estimation error based on the truth are seen to track each other fairly well. They are in the proper order and relative ratios between the two for the various methods are seen to be similar.

It is important to note that contrast trees are not perfect. As with any learning method they can sometimes fail to capture sufficiently complex dependencies on the predictor variables $\mathbf{x}$. In such situations lack-of-fit may be under estimated.

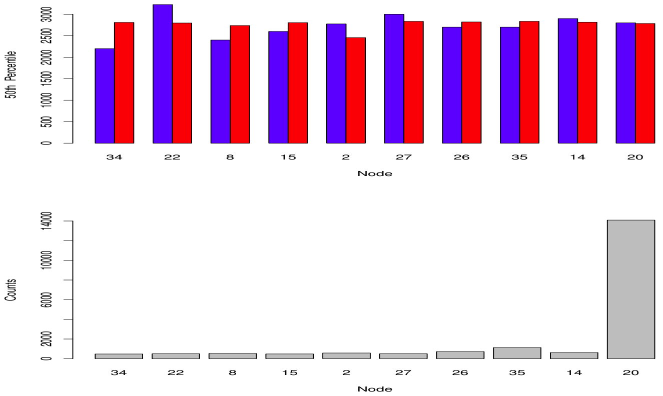

Figure S2: Online news data. Upper frame: Empirical value of the median for observations in each region (blue), along with the corresponding median of the model predictions (red) in that region, for a quantile contrast tree. Lower frame: counts in each region.

Thus contrast trees can reject fit quality but never absolutely insure it.

S2 Quantile regression example

Use of contrast trees in quantile regression is illustrated on the online news popularity data set described in Section 6.3. Here we apply contrast trees to diagnose the accuracy of gradient boosting estimates of the median and 25th percentile function of $y \,|\, \mathbf{x}$.

The usual quantile regression loss used by gradient boosting for estimating the pth quantile z is given by (24) where here $p \in \{0.5, 0.25\}$ and z is the corresponding quantile estimate. With contrast trees we use (7) as a discrepancy measure. This quantity can be interpreted as lack-of-coverage or prediction error on the probability scale. It is the absolute difference between the target probability p and the empirical probability $\Pr(y < z)$ as averaged over the region.

The data were randomly divided into two parts: a learning data set of $N_l = 20000$ and and test data set of $N_t = 19644$. The quantile functions were estimated using the former. A ten region tree to contrast the median of $p_y(y \,|\, \mathbf{x})$ from its gradient boosting predictions was built using (7) on the test data set. The results are shown in Fig. S2.

The upper frame shows the empirical (blue) and predicted (red) median in each of the regions in order of absolute discrepancy (7). The lower frame gives the number of counts in each corresponding region. One sees that for 85% of the data (node 20) gradient boosted model predictions of the median appear to be quite close. In other regions of $\mathbf{x}$ -space there are small to moderate differences.

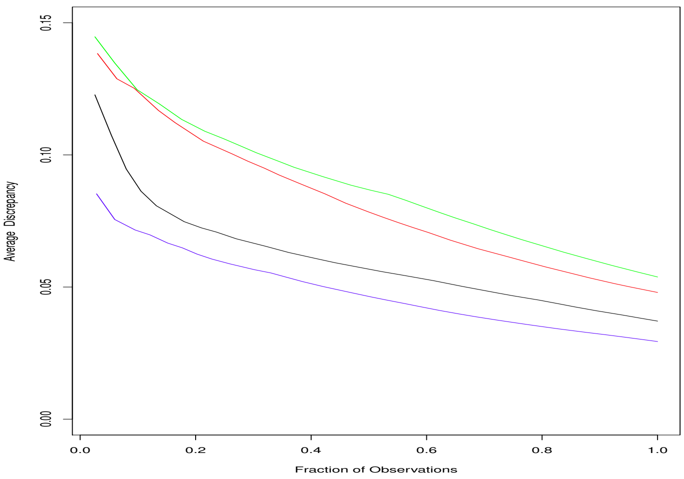

Figure S3: Online news data. Lack-of-fit contrast curves comparing conditional median estimates by constant (green), linear quantile regression (red), gradient boosting (black), and contrast boosting (blue).

Figure S3 shows lack-of-fit contrast curves for estimating the median of y given $\mathbf{x}$ by four methods. The green curve represents a constant prediction of the global median at each $\mathbf{x}$ - value. The red curve is for linear quantile regression. The linear model seems only a little better than the constant one. The black curve represents the gradient boosting predictions based on (24) which are somewhat better. The blue curve is the result of applying contrast boosting (Section 5.1) to the gradient boosting output. Here this strategy appears to substantially improve prediction.

Standard errors for these quantities can be estimated by computing them on repeated bootstrap samples drawn from the data. For the left most points on each curve the bootstraped errors are respectively 0.015, 0.016, 0.018, and 0.016 (top to bottom). For the right most points the corresponding errors are 0.0023, 0.0026, 0.0031 and 0.0033. Thus, the larger differences between the curves seen in Fig. S3 are highly significant.

Figure S4 shows lack-of-fit contrast curves for estimating the conditional first quartile ($p = 0.25$) as a function of $\mathbf{x}$ for the same four methods. Here one sees that the global constant fit appears slightly better that the linear model, while the gradient boosting quantile regression estimate is about twice as accurate. Contrast boosting seems to provide no improvement in this case. Bootstrap standard errors on the left most points of the respective curves are 0.014, 0.0092, 0.020, and 0.015 (top to bottom). For the right most curves the corresponding errors are 0.0027, 0.0039, 0.0029 and 0.0030.

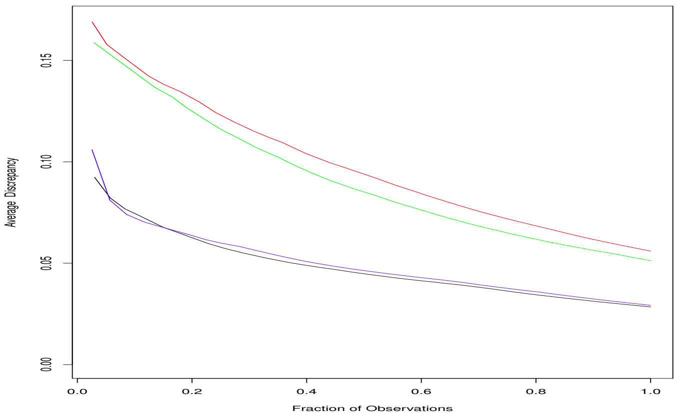

Figure S4: Online news data. Lack-of-fit contrast curves comparing conditional 25–percentile estimates by constant (green), linear quantile regression (red), gradient boosting (black), and contrast boosting (blue).

Table S2

Prediction risk corresponding to the several quantile regression methods for online news data

Method	Median	1st Quartile
Constant	2489.5	678.3
Linear	2488.5	678.4
Gradient Boosting	2481.7	674.1
Contrast Boosting	2479.9	674.1

Table S2 shows quantile regression prediction risk based on L_1 loss (24) for median ($p = 0.5$) and first quartile ($p = 0.25$) using the four methods shown in Figs. S3 and S4. Although here prediction risk measures lack-of-accuracy of the methods in the same order as their respective contrast trees, it gives no indication of their actual relative or absolute lack-of-fit to the data as seen from their respective contrast curves in Figs S3 and S4.

S3 Distribution boosting example

Distribution boosting is illustrated using the online news popularity data described in Section 6.3. The goal is to estimate the distribution $p_y(y \,|\, \mathbf{x})$ of ($\log_{10}$) popularity of news articles y for given sets of predictor variable values $\mathbf{x}$. Here we investigate the variation of the final distribution estimate $\hat{p}_y(y \,|\, \mathbf{x})$ to different initial z - distributions $p_z(z \,|\, \mathbf{x})$. For the same $p_y(y \,|\, \mathbf{x})$, changing the initial $p_z(z \,|\, \mathbf{x})$ distribution can substantially change the nature of the target transformation functions $g_{\mathbf{x}}(z)$ to be estimated. This can affect ultimate accuracy of the estimates $\hat{p}_y(y \,|\, \mathbf{x})$.

Distribution boosting was applied to the 20000 observation training data set using three different initial $p_z(z \,|\, \mathbf{x})$.

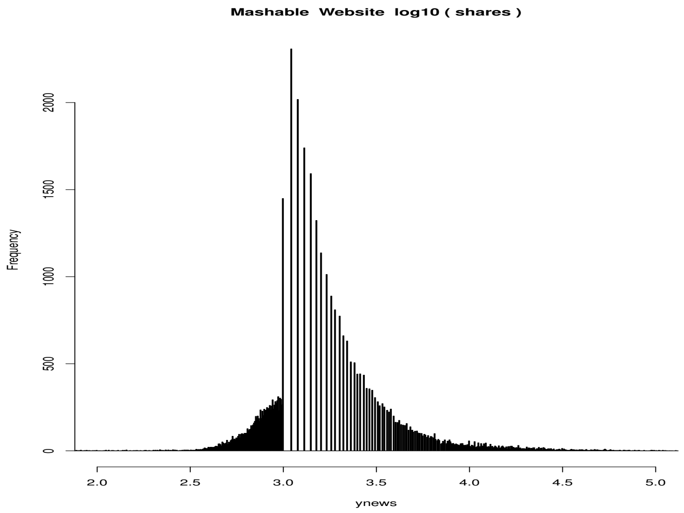

Figure S5: Distribution of $\log_{10}(shares)$ for the online news data.

The first was the same normal distribution $z \sim N(\bar{y}, \sigma_y^2)$ at every $\mathbf{x}$, where $\bar{y}$ and σ_y^2 are the mean and variance of $y = \log_{10}(\text{popularity})$. The second initial distribution is the empirical marginal distribution of y as shown in Fig. S5. This assumes $p_y(y \,|\, \mathbf{x})$ is independent of $\mathbf{x}$. The third initial z - distribution is that of the residual bootstrap at each $\mathbf{x}$ as described in Section 6.3. This assumes homoscedasticity on the log-scale with varying location.

The upper left frame of Fig. S6 shows the distribution of the average pair-wise difference between the three CDF estimates on each (test set) observation $\mathbf{x}$, resulting from the three different beginning z - distributions. Difference between two CDF estimates is given by (22) with the 100 evaluation points $\{u_j\}_1^{100}$ being a uniform grid between the minimum of 0.001 quantiles and the maximum of the 0.999 quantiles of the three distributions.

The 50, 75, and 90 percentiles of the distribution shown in the upper left frame are respectively 0.028, 0.040, and 0.053. As in Fig. 10 the remaining plots in Fig. S6 show the three corresponding CDFs for those observations with pair-wise average difference equal to these respective percentiles. The green curves display the estimate corresponding to the Gaussian starting z - distribution, red for the empirical marginal distribution of Fig. S5, and black for the residual bootstrap start. The upper right frame shows that for at least half of the observations the three estimates are fairy similar. The other half exhibit moderate differences. The residual bootstrap estimates tend to be different from the other two, which are usually similar to each other.

Figure S6 shows that different starting z - distributions give rise to at least slightly different conditional distribution estimates. In general, different methods produce different an-

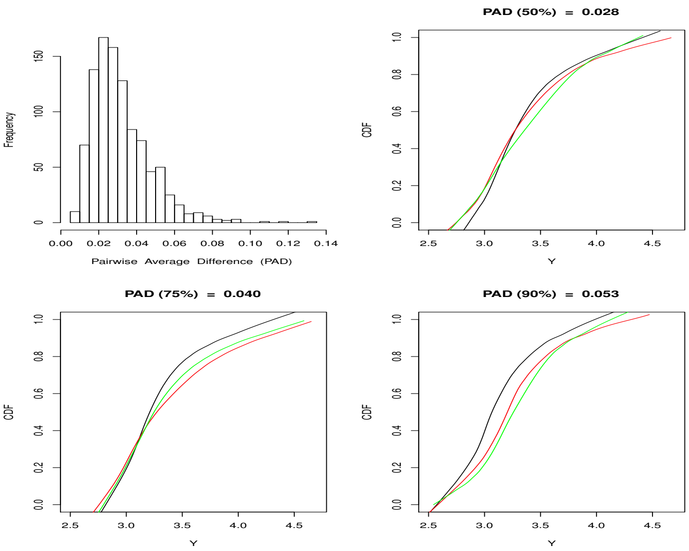

Figure S6: Upper left: distribution of average pair-wise difference between CDF estimates resulting from the three different initial z - distributions for online news data. Upper right: CDF estimates for parametric bootstrap (black), Gaussian (green) and empirical marginal (red) starting distributions for observation with median pairwise difference. Lower: corresponding plots for 75% and 90% decile difference.

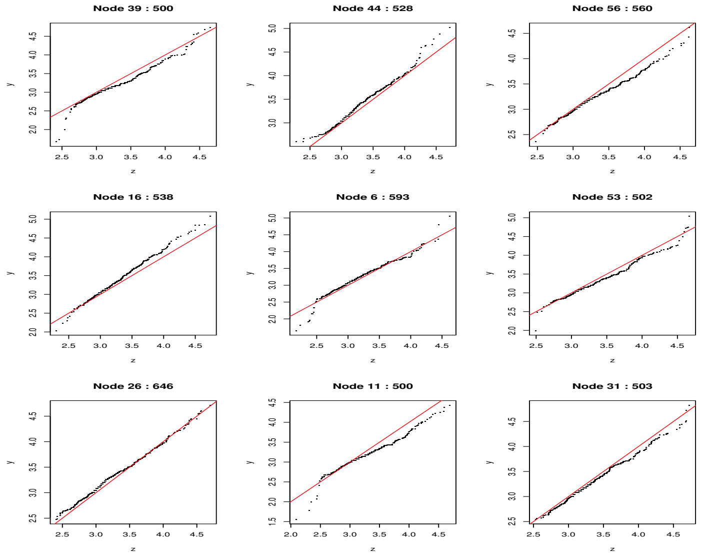

Figure S7: QQ–plots of y versus $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ calculated from parametric bootstrap start for the nine highest discrepancy regions of a 50 node contrast tree on the online news test data set. The red lines represent equality.

swers and one would like to ascertain their respective accuracies. Contrast trees provide a lack-of-fit measure. With conditional distribution estimates one can contrast y with $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ on an independent test data set not involved in the estimation as was illustrated in Fig. 9. Here we employ this strategy to evaluate the respective accuracies of the three conditional distribution estimates obtained by the three different starting z - distributions.

Figure S7 shows QQ – plots of y versus the estimates $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ based on the residual bootstrap starting z - distribution. Shown are the nine largest discrepancy regions of a 50 terminal node contrast tree. These nine regions account for 25% of the data. This can be compared to Fig. 6 which shows the corresponding QQ –plots for y versus the original residual bootstrap z before distribution boosting.

Figure S8 shows the lack-of-fit contrast curves corresponding to the three distribution boosting estimates based on the three different starting z - distributions. Each line summarizes a different tree contrasting y with one of the corresponding three estimates $\hat{y} = \hat{g}_{\mathbf{x}}(z)$. The green and red curves in Fig. S8 summarize the results for contrasting y with $\hat{g}_{\mathbf{x}}(z)$ based on the respective Gaussian and empirical marginal distribution (Fig. S5) starting z - distributions. Their accuracies are seen to be similar. The black curve summarizes the tree depicted in Fig. S7 contrasting y with the estimates $\hat{g}_{\mathbf{x}}(z)$ based on the residual bootstrap starting z - distribution.

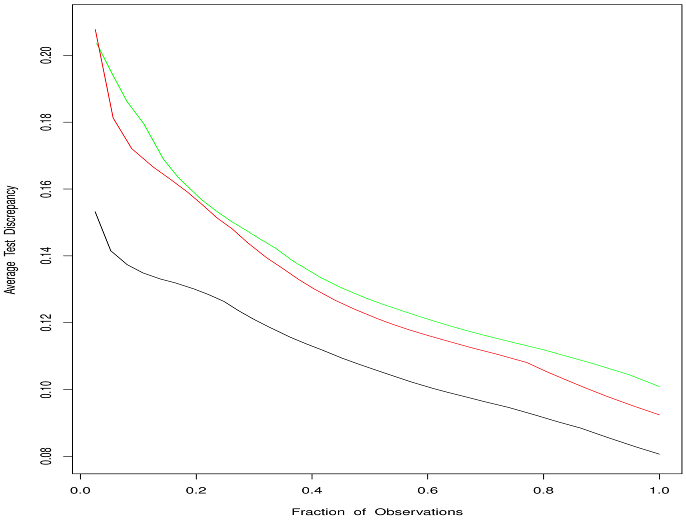

Figure S8: Lack-of-fit contrast curves for three trees contrasting y with $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ based on the different starting z - distributions: Gaussian (green), empirical marginal (red) and parametric bootstrap (black).

These $\hat{g}_{\mathbf{x}}(z)$ estimates appear to be somewhat more accurate. Bootstrap standard errors on the left most points of all three curves are 0.022. For the right most points the corresponding errors are 0.0055, 0.0052 and 0.0049.

The average discrepancy of the tree contrasting y and the residual bootstrap estimated $\hat{g}_{\mathbf{x}}(z)$ (black) is 0.081. The corresponding averages for the respective Gaussian and empirical marginal distribution (Fig. S5) starting z - distributions are 0.10 and 0.092 respectively. These results can be compared with the discrepancies of their initial *untransformed* z - distributions. Average discrepancy for contrasting y with the untransformed residual bootstrap distribution (Fig. 6) is 0.13. The corresponding average discrepancies with y for the untransformed Gaussian z distribution is 0.26, and that for the empirical marginal distribution is 0.24. Thus the residual bootstrap provided a much closer starting point for estimating $p_y(y \,|\, \mathbf{x})$ ultimately resulting in somewhat more accurate results.

One can obtain a null distribution for average transformed discrepancy by repeatedly applying the contrast boosting procedure with y and z randomly sampled from the same distribution. In this case $p_y(y \,|\, \mathbf{x})$ and $p_z(z \,|\, \mathbf{x})$ are the same and any differences detected by the distribution boosting procedure, as revealed by a final contrast tree, are caused by the random nature of the data and not actual differences between the respective distributions. Fifty replications of contrasting boosting based on pairs of random samples, each drawn from from the (same) residual bootstrap distribution, produced and average tree discrepancy of 0.085 with a standard deviation of 0.003. Thus there is no evidence here for a systematic difference between the distribution of the original y and its estimate $\hat{y} = \hat{g}_{\mathbf{x}}(z)$ based on the residual bootstrap initial z - distribution.

S4 Two-sample contrast trees

Contrast trees as so far described are applied to a single data set where each observation has two outcomes y and z, and a single set of predictor variables $\mathbf{x}$. A similar methodology can be applied to two–sample problems where there are separate predictor variable measurements for y and z. Specifically the data consists of two samples $\{\mathbf{x}_i^{(1)}, y_i\}_1^{N_1}$ and $\{\mathbf{x}_i^{(2)}, z_i\}_1^{N_2}$. The predictor values $\mathbf{x}_i^{(1)}$ correspond to outcomes y_i, and the values $\mathbf{x}_i^{(2)}$ correspond to z_i. The goal to identify regions in $\mathbf{x}$ - space where the two conditional distributions $p_y(y \,|\, \mathbf{x})$ and $p_z(z \,|\, \mathbf{x})$, or selected properties of those distributions, most differ.

Discrepancy measures for each region R_m of $\mathbf{x}$ - space can be defined in analogy with (1)

$$d_m = D(\{y_i\}_{\mathbf{x}_i^{(1)} \in R_m}, \{z_i\}_{\mathbf{x}_i^{(2)} \in R_m}). \tag{25}$$

Regions and splits are based on the pooled predictor variable sample $\{\mathbf{x}_i\}_{i=1}^{N} = \{\ \mathbf{x}_i^{(1)}\}_{i=1}^{N_1} \cup \{\mathbf{x}_i^{(2)}\}_{i=1}^{N2}$ with $N = N_1 + N_2$. Tree construction strategy is the same as that described in Section 3 using (25).

We illustrate two–sample contrast trees using the census income data set described in Section 6.1. One of the samples is taken to be the data from the 32650 males, and the other sample data from the 16192 females. The goal is to investigate gender differences in probability of high salary (greater than \$50K/year, \$100K 2020 equivalent) in terms of the other demographic and financial variables as reflected in this data set.

The high salary probability averaged over all males in the data set is 0.30 whereas that for females is 0.11. Thus the relative odds of high salary for men is almost three times that for women over the entire data set. Here we use two–sample contrast trees to investigate whether there are special demographic and/or financial characteristics for which these relative odds are different. Trees were built on a random half sample of 24421 observations and the corresponding node statistics computed on the other left out half.

We first use two–sample contrast trees to seek regions in predictor variable $\mathbf{x}$ - space for which male/female relative high salary probability is larger than 3/1. For this we use a *ratio* discrepancy measure

$$d_m = mean(y_i \,|\, \mathbf{x}_i^{(1)} \in R_m)/mean(z_i \,|\, \mathbf{x}_i^{(2)} \in R_m) \tag{26}$$

where $\{y_i, \mathbf{x}_i^{(1)}\}_{i=1}^{N_m}$ represents the $N_m = 32650$ males and $\{z_i, \mathbf{x}_i^{(2)}\}_{i=1}^{N_f}$ the $N_f = 16192$ females. Here y_i and z_i are indicators of high (male and female) salary and $\mathbf{x}_i^{(1)}, \mathbf{x}_i^{(2)}$ are the corresponding predictor variables.

Figure S9 summarizes results for a ten region contrast tree using (26). In the top frame the height of blue/red bars represent the probability of income greater that \$50K for the women/men in the region. In the bottom frame the blue bars represent the fraction of the 16192 women in the region whereas red signifies the corresponding fraction of the 32650 men. The blue/red horizontal lines represent the female/male global average high salary probabilities.

This contrast tree has found several small regions for which the male/female odds ratio (26) is much greater than its global average of 3/1. For example region 12 containing 551 observations has a 10.3/1 ratio. Region 22 with 501 observations has a 4.6/1 ratio. However, in all of the highest ratio regions the actual male/female probabilities of high salary are well below their respective global averages. In the higher probability regions the ratios roughly correspond to the corresponding global averages.

We next attempt to uncover regions in $\mathbf{x}$ - space where the female/male high salary odds ratio is much greater than its global average of 1/3 by using the inverse discrepancy measure

$$d_m = mean(z_i \,|\, \mathbf{x}_i^{(2)} \in R_m)/mean(y_i \,|\, \mathbf{x}_i^{(1)} \in R_m).$$

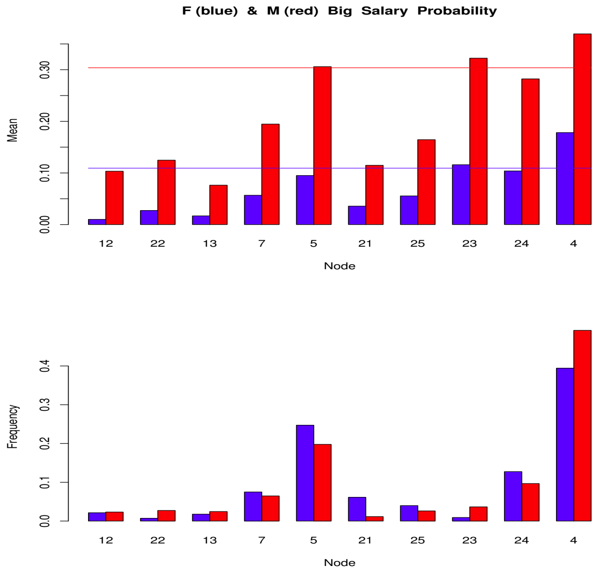

Figure S9: Upper frame: probability of income greater that $50K for women (blue) and men (red) in regions designed for relatively large values of the latter. Lower frame: Fraction of women (blue) and men (red) in each region.

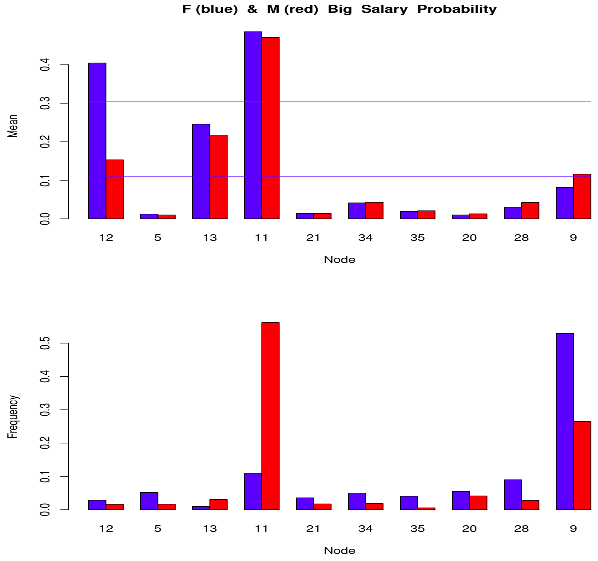

Figure S10: Upper frame: probability of income greater that $50K for women (blue) and men (red) in regions designed for relatively large values of the former. Lower frame: Fraction of women (blue) and men (red) in each region.

Figure S10 summarizes the regions of the corresponding ten region contrast tree in the same format as Fig. S9. The tree has uncovered three regions in which the high salary probability for women is higher than that for men and much higher than its global average (blue line). In region 12 the female/male high salary odds ratio is 2.6/1. In regions 13 and 11 the probabilities are about equal. In region 11 the overall probability of high salary for both is relatively very high (0.47). This region contains 57% of the males and only 11% of the females in the data set. The rules defining these three regions are

Node 12

$22 \leq \text{age} < 50$

&

martial status = never married

&

hours/week ≤ 34

Node 13

age > 50

&

martial status = never married

&

hours/week ≤ 34

Node 11

age > 22

&

martial status = never married

&

hours/week > 34

This data set was originally constructed for the purpose of comparing performance of various machine learning algorithms for predicting high salary. There is no information as to its representativeness, even for 1994. The analysis presented here is meant to illustrate the variety of the types of problems to which contrast trees can be applied.

Contrast trees can be used to compare any two samples based on the same measured quantities. In particular, the two samples may be taken from the same system under different conditions or at different times. In these situations contrast trees can detect the presence of any associated "concept drift" between the samples, and if detected explain its nature.