LECTURE 15: Local Data Mining: Trees, and Rule-Based Classification and Decision-Making

"Trees" (1913)

I think that I shall never see
A poem lovely as a tree.

A tree whose hungry mouth is prest
Against the earth's sweet flowing breast;

A tree that looks at God all day,
And lifts her leafy arms to pray;

A tree that may in Summer wear
A nest of robins in her hair;

Upon whose bosom snow has lain;
Who intimately lives with rain.

Poems are made by fools like me,
But only God can make a tree.

A tree model is the antithesis of the neural net model. In contrast to the neural net using **global** information to categorize data in a black box hard to understand way, the tree model uses **local** information to categorize data in a simple way way.

A tree model breaks up data on, say a person's attributes that predict earnings one factor at a time – say by gender into branches M and F and then breaks each branch by another factor – say age **separately** along the gender branch without consideration of how that factor affects outcomes on a different branch. It you know that age increases earnings for men, the tree model ignores that in assessing how age affects female earnings. **It predicts locally.**

The neural net adjusts coefficients for every factor to make its prediction. An observation comes in for a man and it adjusts predictions for the entire data set, including all women and for all other factors **It predicts globally.**

Compare the two machine-learning algorithms to the standard labor economics approach.

Standard human capital/ln wage approach: ln wage = a +b Yrs Schooling + c Age + d $(Age)^2$ + e Gender, where analyst specified factors based on potential impact on productivity/discrimination/etc Could add interaction terms say of Schooling with Gender

Neural net predicts wages with non-linear equation with interactions built into layers of nodes /logistic form.

Trees predict by dividing sample into pairs for each set of factors, builds interactions in naturally for each division by allowing different effects of other variables on branch but ignores effects common to groups .

Standard regression usually fit to all the data (putting data-driven choices under rug). Both neural net and tree face more serious over-fitting problem in that they estimated to fit data and could have made "over-fitting" error.

Formally a tree is a data mining model that classifies objects into categories using a non-linear interaction **without any distribution assumptions.** Trees are suited for classifying people or predicting membership in a group – Who listens to Hillbilly at Harvard on WHRB? The tree is a **binary recursive partition.** It splits parent nodes into two groups and continues until it reaches some stop. .Goes from roots to terminal leafs. Each step of the partition is the best separation between the classes according to a function that specifies cost of mis-classification.

Tree models are robust and widely used. Main tree algorithms are:
CART, http://www.salford-systems.com/products-cart.html;
C5.0 www.rulequest.com/;

AnswerTree (SPSS) http://www.spss.com/answertree/index.htm – which includes QUEST, CHAID, CART.
SAS Enterprise Miner http://support.sas.com/documentation/whatsnew/91x/emguiwhatsnew900.htm.

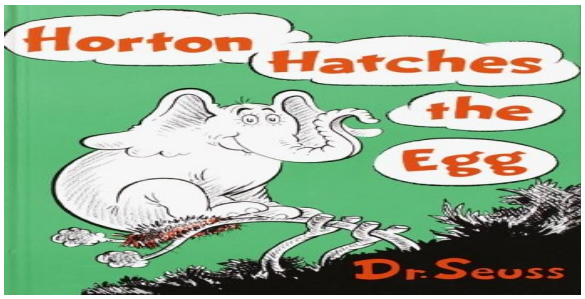
Excel simple tree model --http://www.geocities.com/adotsaha/CTree/CtreeDemo.html by Indian engineer;

Stata has tree models as well – https://www.stata.com/meeting/spain15/abstracts/materials/spain15_mora.pdf

R has tree model also.

DTREG Software For Predictive Modeling and Forecasting – Neural nets, tree based methods Can download limited version free FOR 30 DAYS. www.dtreg.com/download, Contains trees + neural net + ...

Tree models generate rules that make sense and can be displayed graphically. They produce RULES like: IF HEIGHT> X and>WEIGHT> 800 and BIG EARS and SITS ON EGG FOR WEEKS, THIS IS HORTON THE ELEPHANT.



Algorithm: 1.splits each node in a tree;
2.decides when a tree is complete; and
3.assigns an outcome to each terminal node

Random forest goes beyond single tree to build many tree models by randomly selecting DIFFERENT **training** data to create a forest of models and then has the models vote to classify an object. Each model takes data from a new observation and says where to place the observation. The forest chooses classification with the most votes.

Tree model vs regression model: to predict criminal banker: Age, Ethnicity, Sex, FamInc, Wall Street #, University.

Standard linear or logit 0/1 regression of Criminal on Xs uses **ALL THE INFORMATION TO FIT EVERY PARAMETER** –

IT IS GLOBAL SEARCH. (Logit: **P = 1/[1+ exp (-Bx)]** so that biggest value is 1 and smallest is 0; linear discriminant P = f(X).Similarly, neural net model uses every observation to develop weights on links/edges

Tree model USES LOCAL DATA. Once data is partitioned into two sub-samples at the root node, tree model analyzes each sub sample separately. As the partitioning continues, analysis conducted within group so that discovery of patterns becomes more local. Information from different nodes is not pooled or combined, **the "fit" at one node is never adjusted to take into account the fit at another node.**

HOW IS THIS RELATED TO CELLULAR AUTOMATA MODELS?
Do doctors use local or global information to diagnose patients?

For example if we split by gender first, and we find young men more likely to criminal bankers than older men, we do not use this age relation to judge women, where perhaps older women are more likely to be criminal bankers.

TREE MODELS	**LOGIT/REGRESSION MODELS**
Automatic analysis – fast model	Requires statistical model
Surrogates for missing values	Discards missing variables or or uses means + dummy
Unaffected by outliers	Affected by outliers
Handles non-numeric data readily, but must bin numeric	Does numeric well, dummies for non-numeric
No dimension problems	Limited number of regressors
Splits usually by single variable--> interactions	**Must specify interactions**
Discontinuous response	Continuous change
Small change in *x* could lead to a large change in *y*	Small change in x usually --> small change in y
Coarse-grained –output is step function	Fine-grained predictions
A 17-node tree can only predict 17 different outcomes	Each observation has separate prediction

KEY TREE ISSUE: HOW TO MAKE SPLITS to get observations into homogeneous classes or subsets.

At each step split the tree (usually in two ways, though could do more) to **minimize the squared deviation of predictions from actual** (could use another metric for goodness of model).

For example, say you have the following data on gender, handedness (aka chiralilty), and height

Gender	Chirality	Height
M	R	1.0
F	R	. 6
M	L	.9
F	R	.7
F	L	.8

The mean of Height is 0.8
Variance for all the observations is $0.10 = (0.2)^2 + (0.1)^2 + (0.1)^2 \ (0.2)^2$

You want a split that to reduce the variance. Divide along biggest split first. Mean **MF gap is 0.25** with variance of $(2(.05)^2$ for M and $2(.1)^2$ for F) or .025 as total variance . Mean RL split **is .08** (= 0.85 for L vs 0.77 for R) with variance of $2(.05)^2 = .0025$ for L and $(.23)^2 + .07^2 + .17^2 = .087$.

So first split M F, then go to LR. In the data you can see L is highest for F and lowest for M so that better to treat LR as separate within M and F. Variance in predicted or dependent variable declines. The further along the tree the less variability in the dependent variable. Splits send cases with x<c to left and x>c to right. Variance in predictor variables also drastically reduced.

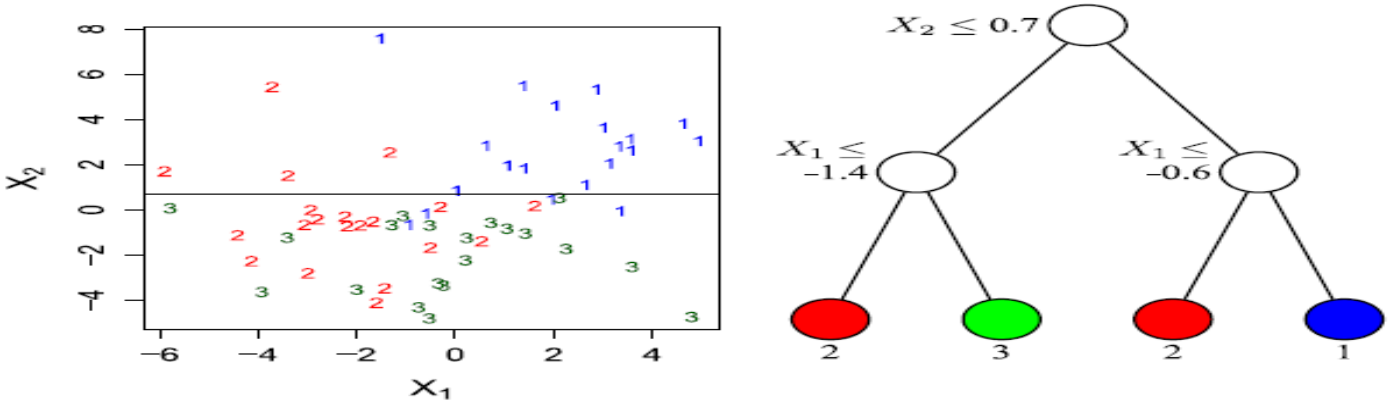


Figure 1: Partitions (left) and decision tree structure (right) for a CART classification model with three classes labeled 1, 2, 3. At each intermediate node, a case goes to the left child node if and only if the condition is satisfied. The predicted class is given beneath each leaf node.

The tree is a classifier *f*(**x**, **w**): **x** –> y that assigns a branch/label to an object with feature vector x on **training set.**

It classifies the observations in node *m* to the major class in the node: P_{mk} is the proportion of observation of class k in node m. The model seeks to minimize % misclassified. To generalize, it tests classifier by moving to a different body of data **(test set),** as shown in critical graph below:

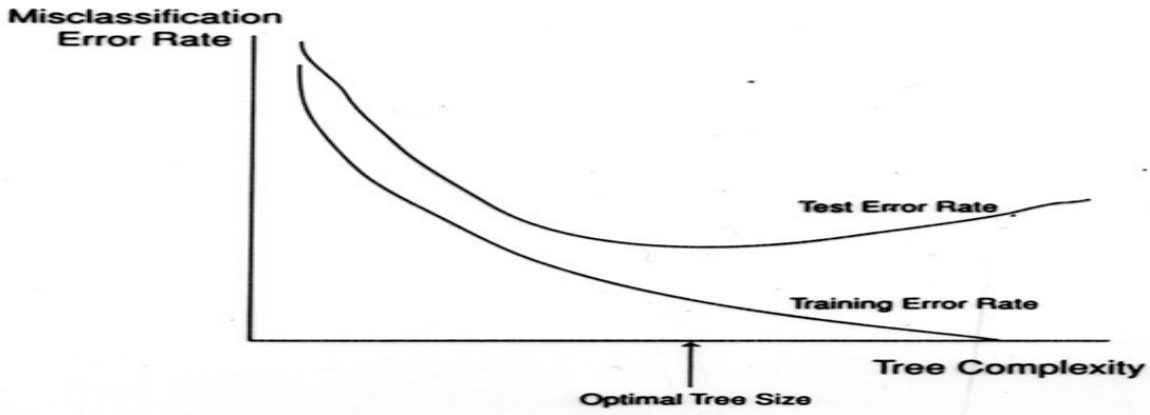


Figure 5.4 A hypothetical plot of misclassification error rates for both training and test data as a function of tree complexity (e.g., number of leaves in the tree).

The training error declines as you add more nodes/leaves but the test error rises after the point where you have over-fit the data, giving the optimal tree size or model. So to avoid over-fitting fit tree on a training set, then test it on a 2nd data set. If not enough data, use "bootstrap cross-validation": randomly remove 1/10th of data and fit on 9/10ths of the data. Then use 1/10th to test. Do this again and again.

Other ways to deal with over-fitting: PRUNING A TREE; PRICING NODES

Grow a large tree with many nodes. Then eliminate parts of tree not supported by test data: "CART's developers determined ... that no stopping rule could be relied on to discover the optimal tree, so they introduced the notion of over-growing trees and then pruning back … (this) ensures that important structure is not overlooked by stopping too soon."

Another procedure is to attach a price to # of nodes of the tree and minimize # Misclassified Observations (a function of #Nodes + "price" of complexity (# Nodes). Since increasing # nodes reduces the # misclassified but raises the cost of complexity, this model balances cost of misclassification vs cost of complexity: MB = MC.

Another is to minimize Akaike information criterion (AIC), ln form that rewards goodness of fit, but penalizes the number of estimated parameters. (With k variables and n observations and likelihood measure of fit of the model lnL, the AIC for

model c is AICc = 2k-2 lnL(k) (as n--> infinity). As you add variables, better fit lowers cost by (-2 ln L (v)) but pay 2k penalty.)
David Donoho (Stanford stats) suggests that best penalty is 2 σ^2 log(D), where D is # of nodes/ dimensions/ measures/variables:"with this logarithmic penalty, one can mine data to one's taste while controlling the risk of finding spurious correlation." (http://www-stat.stanford.edu/~donoho/Lectures/AMS2000/Curses.pdf)

TREE MODELS ARE RULES: Each node is a rule for placing observations into classes according to criterion. Consider two classes y = {0, 1} with the loss function $L(y, \boldsymbol{f}(\mathbf{x}, \omega)) = 0$ if $y = \boldsymbol{f}(\mathbf{x}, \omega)$ – no error if correct; error of 1 if $y \neq \boldsymbol{f}(\mathbf{x}, \omega)$ ie incorrect. The goal is to find rule $f(x, w)$ that minimizes future classification error in the future.

Can adjust misclassification penalties to avoid the most costly errors by specifying higher penalty for misclassifying key data: "CART can accommodate situations in which some mis-classifications, or cases that have been incorrectly classified, are more serious than others." Can specify different s**plitting criteria:** CART includes .. Gini, symmetric Gini, twoing, ordered twoing, class probability for classification trees, and least squares and least absolute deviation for regression trees - and one multi-variable splitting criteria, the linear combinations method. The default Gini typically performs best, but, given specific circumstances, other methods can generate more accurate models.

IF TREES WORK, GROW A FOREST?

Take a random sample of observations, fit one tree. Do it again and again. You will get moderately different rules for classifying. For each new observation you take a vote of the trees and classify according to majority vote. Instead of training/test set, you **use the different trees to avoid over-fitting**. Think of the Forest as using global data to generalize trees based on local search. Averaging among decision trees improves matters because trees are LOCAL learners, so that models based on different samples from the data give different local approximations. By averaging several decision tree models, you average out the random noise that produces over-fitting since the models over-fit different parts of the data. **NB: Averaging models differs from using one big model that averages coefficients**.

Theorems prove that for a large # of trees, random forests do not over-fit but produce a limiting value of the generalization error. A tree model that over-fits one data set will be "outvoted" by models from other parts of data space. Mis-classification error is smaller when each tree has low error (predicts well) and when the trees are not highly correlated, so variety in the forest is good. The smaller correlation between the trees/the square of the ability of the trees to predict better is the Forest predictor. http://stat-www.berkeley.edu/users/breiman/RandomForests/cchome.htm).

FORESTS DO BETTER THAN TREES IN SEVERAL EXAMPLES (http://www.stat.berkeley.edut-tice/scma-breiman pdf). Here are 10 examples

Table 1. Data Set Descriptions.

Data Set	Training	Test	Variables	Classes
cancer	690	-	9	2
ionosphere	351	-	31	2
diabetes	768	-	8	9
glass q18M	214	-	9	6
soybean	683	-	35	19
Team-	15,000	5000	16	26
satellite	4,435	2000	36	6
shuttle	43,500	14,500	9	7
DNA	2,000	1,186	60	3
digit	7,291	2,007	256	10

Table 2. Test Set 1 misclassification Error (%)

Data Set	Forest	Single Tree
Breast cancer	2.9	5.9
ionosphere	5.5	1 1.2
diabetes	24.2	25.3
glass	22.0	30.4
soybean	5.7	8.6
letters	3.4	12.4
satellite	8.6	14.8
shuttle	7.0	62.0
DNA	3.9	6.2
digit	6.2	17.1

For the five smaller data sets above the line, the test set error was estimated by leaving out a random 10% of the data, then running CART and the forest on the other 90%. And doing this lots of time. Bootstrap technique

Some random forest model builds trees that select a small group of input variables to split at random at each node. Why would you want to split at random rather than splitting to minimize variance due to split**?**

EXAMPLES OF TREE MODELS

1. The Application of a Classification-Tree Model for Predicting Low Back Pain Prevalence Among Hospital Staff Mendelek, et al, Archives of Environmental & Occupational Health, Vol. 68, No. 3, 2013

ABSTRACT. Low back pain (LBP) is a widespread musculoskeletal condition that frequently occurs in the working-age population (including hospital staff). This study proposes a classification-tree model to predict LBP risk levels in Sacre-Coeur Hospital, Lebanon … using … individual and occupational factors. T**he developed tree model explained 80% of variance in LBP risk The proposed tree model can be used by expert physicians in their decision-making for LBP diagnosis among hospital staff.**

2. Predicting rare event: "Comparing Random Forest with Logistic Regression for Predicting Class-Imbalanced Civil War Onset Data" (Muchlinski,Siroky He,Kocher Political Analysis (Winter 2016) 24 (1): 87-103

The most commonly used statistical models of civil war onset fail to correctly predict most occurrences of this rare event in out-of-sample data. .. We compare the performance of Random Forests with three versions of logistic regression (classic logistic regression, Firth rare events logistic regression, and L1-regularized logistic regression), and find that **the algorithmic approach provides significantly more accurate predictions of civil war onset in out-of-sample data** than any of the logistic regression models.

The Civil War Data (CWD) are measured annually for each recognized country in the world from 1945 to 2000 (Hegre and Sambanis 2006). The dependent variable is a binary measure of whether a civil war onset occurred for a given country, i, in a given year j. N is equal to 7141 county-years. X is a matrix of eighty-eight predictor variables. Using ten-fold cross-validation, we trained our models on nine of the ten folds of the data, then passed the predictions made in the training sets to the test data.

We updated the CWD for all countries in Africa and the Middle East from 2001 to 2014. The updated data give us an additional 737 observations with twenty-one civil war onsets. We trained each model or algorithm on the 1945–2000 CWD and tested on the updated CWD for Africa and the Middle East. A**ll logistic regression models fail to specify any civil war onset in the out-of-sample data. Random Forests correctly predicts nine of twenty civil war onsets in this out-of-sample data when the threshold for positive prediction is 0.50.** Random Forests correctly predicts the onset of civil war in Iraq, Somalia, the Democratic Republic of the Congo, Uganda, Rwanda, and Liberia. It fails to correctly predict the civil wars resulting from the U.S. invasion of Afghanistan, or the civil wars in Syria and Libya that resulted from the Arab Spring. See https://ourworldindata.org/civil-wars

Table 1 Predicted probability of civil war onset: Logistic Regression and Random Forests

	Models and predicted probability of civil war onset			
Civil war onset	*Fearon and Laitin (2003)*	*Collier and Hoeffler (2004)*	*Hegre and Sambanis (2006)*	*Random Forests*
Afghanistan 2001	0.01	0.01	0.01	0.09
Angola 2001	0.04	0.01	0.01	0.13
Burundi 2001	0.00	0.00	0.00	0.05
Guinea 2001	0.00	0.00	0.01	0.22
Rwanda 2001	0.02	0.00	0.00	0.56
Uganda 2002	0.03	0.05	0.00	0.81
Liberia 2003	0.01	0.03	0.00	0.94
Iraq 2004	0.04	0.01	0.00	0.68
Uganda 2004	0.02	0.01	0.02	0.52
Afghanistan 2005	0.01	0.02	0.01	0.14
Chad 2006	0.01	0.07	0.02	0.21
Somalia 2007	0.00	0.00	0.00	0.52
Rwanda 2009	0.00	0.01	0.00	0.74
Libya 2011	0.00	0.01	0.00	0.34
Syria 2012	0.00	0.04	0.00	0.25
DR Congo 2013	0.00	0.00	0.00	0.76
Iraq 2013	0.01	0.00	0.00	0.25
Nigeria 2013	0.01	0.00	0.00	0.25
Somalia 2014	0.01	0.04	0.01	0.87

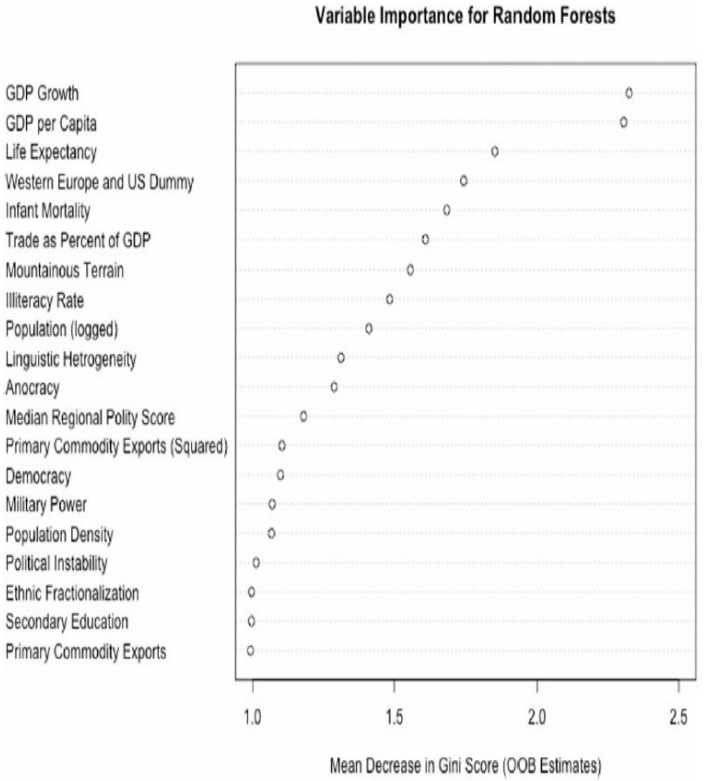


Fig. 4 Plot of variable importance by mean decrease in Gini Score.

3. Decision tree and random forest models for outcome prediction in antibody incompatible kidney transplantation (Shaikhina et al) **Biomedical Signal Processing and Control** (2017)

Clinical data sets are commonly limited in size, thus restraining applications of Machine Learning (ML)techniques for predictive modeling... We explored the potential of Decision Tree (DT) and Random Forest (RF) classification models, in the context of small data set of 80 samples, for outcome prediction in high-risk kidney transplantation. The DT and RF models identified the key risk factors associated with acute rejection: the levels of the donor specific IgG anti-bodies, the levels of IgG4 subclass and the number of human leucocyte antigen mismatches between the donor and recipient. Furthermore, the DT model determined dangerous levels of donor-specific IgG subclass antibodies, thus demonstrating the potential of discovering new properties in the data when traditional statistical tools are unable to capture them. The DT and RF classifiers developed in this work predicted early

transplant rejection with accuracy of 85%, thus offering an accurate decision support tool for doctors tasked with predicting outcomes of kidney transplantation in advance of the clinical intervention.

Table 1
Predictive performance of the two ML models.

Performance measures as defined in Section 2.4	DT		RF	
	training	test	training	test
Correct classification rate, *C* (%)	85.0	85.0	91.7	85.0
Sensitivity, *Sn* (%)	85.7	81.8	93.9	92.3
Specificity, *Sp* (%)	84.0	88.9	88.9	71.4
Positive Predictive Value, *PPV* (%)	88.2	90.0	91.2	85.7
Negative Predictive Value, *NPV* (%)	80.8	80.0	92.3	83.3
Area under the ROC curve, *AUC*	0.849	0.854	0.914	0.819

4. Trees vs Neurons: Comparison between random forest and ANN for high-resolution prediction of building energy consumption (M.Waseem et al) **Energy and Buildings 147 (2017) 77–89** Energy prediction models are used in buildings as a performance evaluation engine in advanced control and optimisation, and in making informed decisions by facility managers and utilities for enhanced energy efficiency.We compared the performance of feed-forward back-propagation artificial neural network (ANN) with **random forest** (RF), an ensemble-based method gaining popularity in prediction – for predicting the hourly HVAC energy consumption of a hotel in Madrid, Spain. Incorporating social parameters such as the numbers of guests marginally increased prediction accuracy in both cases. Overall, **ANN performed marginally better than RF** with root-mean-square error (RMSE) of 4.97 vs 6.10..,. However, the ease of tuning and modeling with categorical variables offers ensemble-based algorithms an advantage for dealing with multi-dimensional complex data, typical in buildings. RF performs internal cross-validation(i.e. using out-of-bag samples) and only has a few tuning parameters.

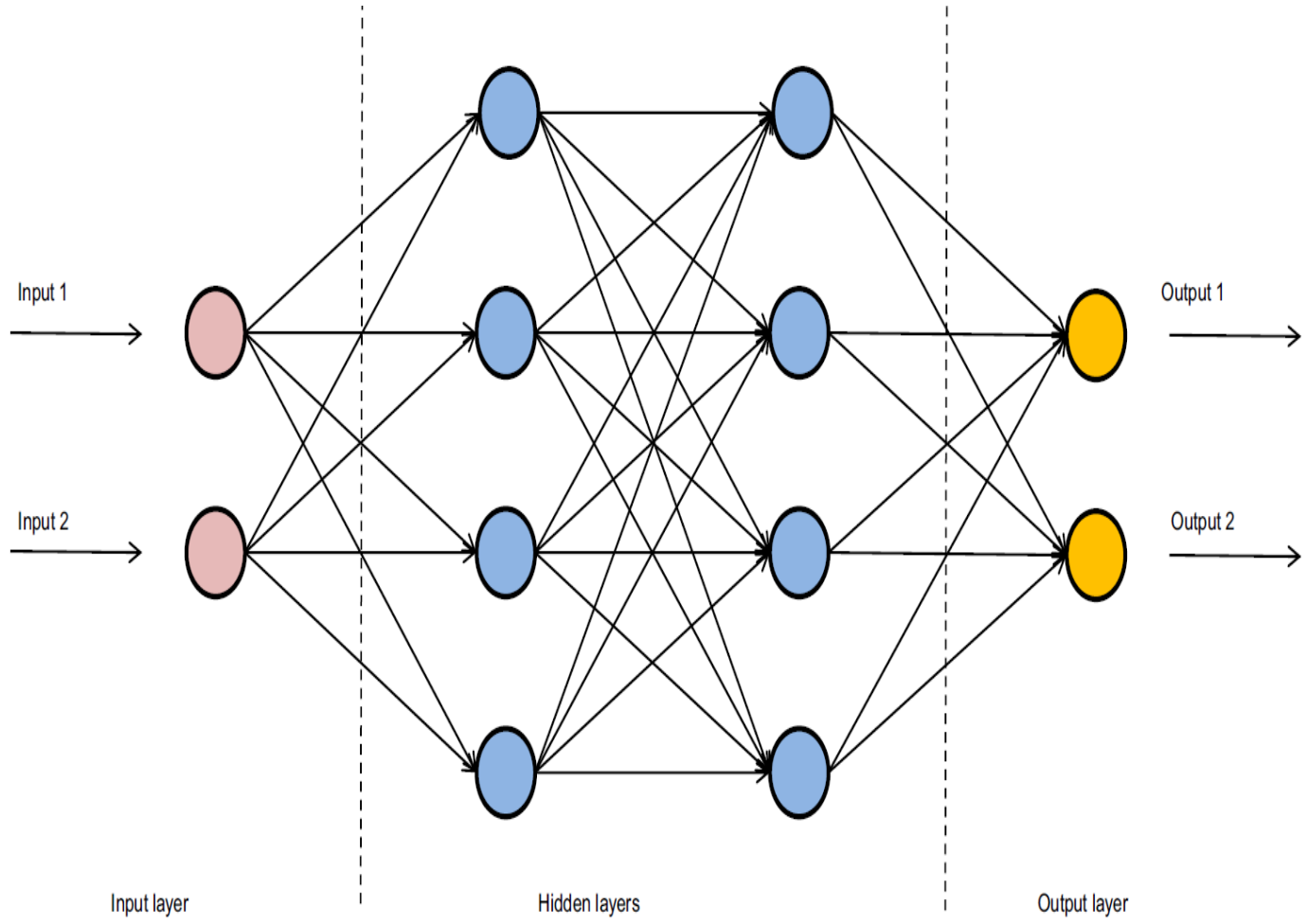


Fig. 2. Schematic diagram of a feed-forward artificial neural network.

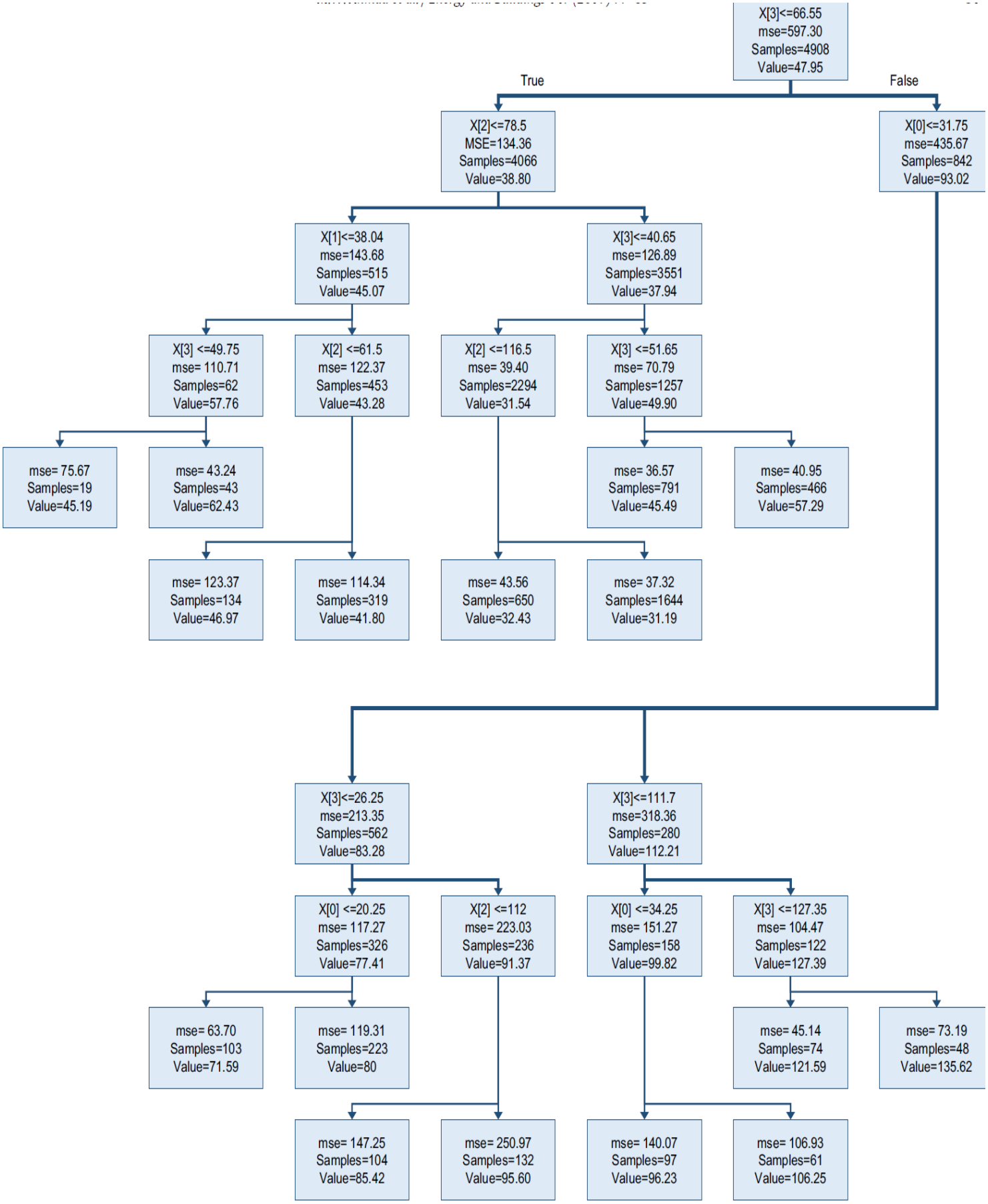


Fig. 3. Decision tree from a random forest for predicting Hotel's HVAC energy consumption. Note: *X*[0]: outdoor air temperature, *X*[1]: relative humidity, *X*[2]: number of rooms booked, *X*[3]: previous value of E.C.