1. Introducing “Classification and Regression Trees” (DTA-CART)

Introducing Decision Theory Analysis (DTA) and Classification and Regression Trees (CART)

John J. McArdle
University of Southern California
Fall 2013

Overview
1. Introducing Decision Theory Analysis (DTA) or Classification and Regression Trees (CART)
2. Using Logistic Regression Logic
3. Considering Binary Classification Trees
4. Using Linear Regression Logic
5. Considering Regression Trees
6. Using Classification Analyses
7. Considering Unsupervised CART
8. Introducing SEM Trees
9. Summary of DTA/CART Research

Introducing CART
• The typical multiple regression prediction alternatives (e.g., Hierarchical, Stepwise, Best Subsets) represent classical way to accomplish the basic analytic goals of dealing with multiple predictors.
• Classification and Regression Tree (CART) is a relatively newer form of data analysis that is “computer-assisted” and has been formally developed over the past 3 decades.
• Some of the popularity of CART comes from its topographic Trees and Cartesian subplots.
• CART programs are now widely available and very easy to use and interpret.
**Computer Issues of Sonquist (1970)**

This investigation had its origin in what, in retrospect, was a rather remarkable conversation between Professor James Morgan, the author, and several others, in which the topic was whether a computer could ever replace the research analyst himself, as well as replacing many of his statistical clerks.

Discarding as irrelevant whether or not the computer could "think," we explored the question whether or not it might simply be programmed to make some of the decisions ordinarily made by the scientist in the course of handling a typical analysis problem, as well as doing computations.

This required examining decision points, alternative courses of action, and the logic for choosing one rather than the other; then formalizing the decision procedure and programming it, but with the capacity to handle many variables instead of only a few.

---

**CART Motivation (from Ritschard, 2007)**

"Particularly in the social sciences, there are two powerful reasons for believing that it is a mistake to assume that the various influences are additive. In the first place, there are already many instances known of powerful interaction effects — advanced education helps a man more than it does a woman when it comes to making money. [...] Second, the measured classifications are only proxies for more than one construct. [...] We may have interaction effects not because the world is full of interactions, but because our variables have to interact to produce the theoretical constructs that really matter." (Morgan and Sonquist, 1963, p 416.)

"Whenever these interrelationships become very complex—containing non linearity and interaction—the usefulness of classical approaches is limited." (Press et al., 1969, p 364.)

---

**History of DTA/CART**

- 1959: Belson suggests that matching can be prediction and creates binary search strategy (see Fielding & O’Muircheartaigh, 1977).


- Later they created popular CHIAD and THIAD programs. It seems as if they were trying to "automatically mimic scientists at work."

- A large number of other scientists joined in these efforts (e.g., ELISEE and IDEA; see Ritschard, 2007)

---

**History of DTA/CART**

- 1980-1990: Berkeley/Stanford Statistics Department revives this kind of work. Leo Breiman (1984) develops the original CART program and Jerome Friedman develops MARS.

- 1990-2000: Although there is much resistance to CART/MARS, there are the seeds of growth in respectability among many statisticians (many pubs).

- 2001-2009: Growth of many available package programs and applications. Most popular in Business research, but also in Molecular genetics research.

- 2009: Psychologists hold first APA-sponsored DTA/CART conference at USC!
First CART from Sonquist & Morgan (1964)

Understanding of Tree Expectations

Understanding Tree Differences

CLASSIFICATION AND REGRESSION TREES
As an example, consider the decision tree generated for heart attack patients at a major teaching hospital (Figure 1 below). The medical staff needed to accurately determine whether a patient was at high risk of a second lethal heart attack as quickly and simply as possible. A high-risk patient would be placed on an intensive care unit for constant monitoring while a low-risk patient could remain on a standard medical unit.

The data for the analysis consisted of the medical records of 219 patients. Each record contained 19 measures taken within the first 24 hours following admission, including basic variables such as blood pressure, age, and presence of tachycardia, and more elaborate measures such as enzyme concentrations based on blood work. Within 30 days following admission, 37 patients died (the high-risk group) while 178 survived (the low-risk group) (see Gilpin, Ohbren, Henning and Ross, 1983).

Based on a systematic analysis of these data, CART produced the following classification scheme, drawn in the form of an inverted tree and read like a flow chart.
Superb Technical Discussion of Tree Splits

Development of Automatic Cross-Validation Procedures (Train → Test)

The famous SPAM filter as a tree (Hastie et al., 2001)

Resulting Indicators of Fit for SPAM

In medical classification problems, the terms sensitivity and specificity are used to characterize a test. They are defined as follows.

Sensitivity: probability of predicting disease given true disease.
Specificity: probability of predicting non-disease given true non-disease.

We define a spam and not spam, or presence and absence of disease, respectively, then Table 6.3.3 we have:

<table>
<thead>
<tr>
<th></th>
<th>Positive</th>
<th>Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not spam</td>
<td>Sensitivity</td>
<td>Specificity</td>
</tr>
<tr>
<td>Spam</td>
<td>0.85</td>
<td>0.92</td>
</tr>
</tbody>
</table>

In this example in order to obtain better results, we define a fixed value of $L_0$ in the two assessed with predicting, a selected choice of $L_0$. By varying the volumes of the boxes $L_0$ and $L_0$, we increase the specificity and decrease the sensitivity of the box or vice versa. In this example, we want to avoid making good results in spam, and therefore we want specificity to be very high. We can achieve this by setting $L_0 = 1$, and $L_0 = 1$. The figure plots the classification and decision tree class 1 (spam) if the proportion of spam is $\geq 0.5$. $L_0$, $L_0$, and class are observable. The
**Important Graphical Contributions**

---

**Key Behavioral Science Applications (Monahan et al., 2001)**

---

**More Behavioral Science Risk Factors**

---

**Note:** Although the topography is very similar, DTA/CART differs from both Neural Networks and Path Analysis
Available DTA/CART Computer Programs

- **R-Code** – many state-of-the-art algorithms that are general and freely available (e.g., Party, RPART, RATTLE, TrajMINR, MASS, and SEMtree).
- **XLMINER** – very popular and inexpensive suite of programs built around on EXCEL structures (also see XLFIT, TABU, and RTREE).
- **SAS Enterprise, SPSS Clementine, SAS SEARCH, and SYSTAT Trees** – added EDM suites for well known packages of programs with broad and general use.
- **SPM=CART, MARS, TREENET** — excellent specialized software, but relatively expensive.
- **SEM based programs marketed as SEM rather than CART** – SAS NL MIXED, PLS, WinBUGS, and Latent GOLD.

2. Binary Outcomes with Logistic Regression

**When do we need special binary outcome regression models?**

In a linear model expressed (for n=1 to N) as

\[ Y_n = \beta_0 + \beta_1 X_n + e_n \]

where the outcome \( Y \) is binary \([0,1]\), and the marginal mean is the proportion \( P \), so:

1. The predicted score may fall outside the reasonable range of 0 to 1 and not “normal”
2. The variance of the prediction will not be homoskedastic: \( \Sigma(Y) = P \) and \( \Sigma(VY) = P(1-P) \)
3. The appropriate distribution of the error will not be \( N(0, \Sigma) \) under this representation

So nothing new is needed when \( P \approx 0.5 \)

**Logistic regression models**

- We write a logit model expressed (for n=1 to N) as

\[ \ln \{ P(g) / (1-P(g)) \} = \beta_0 + \beta_1 * X(g) \]

where the outcome \( Y \) is binary \([0,1]\), \( g=1 \) to \( G \) common groups, and the “odds” = \( P(g) / (1-P(g)) \)

- This model can be rewritten in terms of odds as

\[ \{ P(g) / (1-P(g)) \} = \exp(\beta_0 + \beta_1 * X(g)) \]

or in terms of the probability as

\[ P(g) = \exp(\beta_0 + \beta_1 * X(g)) / (1 + \exp(\beta_0 + \beta_1 * X(g))) \]

- This model can be fit using MLE, and is evaluated using standard likelihood ratio tests and “pseudo \( R^2 \)” (see McCallaugh, 1987).
Logistic regression of $Y = \text{fun}(X)$

SAS input for “Heart Disease”

```sas
FILENAME chd_hl 'c:\data\chd_hl.dat';
PROC LOGISTIC
   DATA=chddata DESCENDING ORDER=DATA OUTEST=outest2;
   MODEL chd = age / RSQUARE CTABLE PPROB=.50 LINK=logit;
   OUTPUT OUT   = outpred2
                  P     = probhat
                  LOWER = problcl
                  UPPER = probucl;
RUN;
TITLE2 'Printing out individual logistic regression estimates';
PROC PRINT DATA=outpred2;
   VAR chd age probhat problcl probucl;
RUN;
```

Logistic results on “Heart Disease”

<table>
<thead>
<tr>
<th>Ordered Value</th>
<th>chd</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>43</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>57</td>
</tr>
</tbody>
</table>

Probability modeled is chd=1.

R-Square 0.2541 Max-rescaled R-Square 0.3410

Analysis of Maximum Likelihood Estimates

<table>
<thead>
<tr>
<th>Parameter</th>
<th>DF</th>
<th>Estimate</th>
<th>Error</th>
<th>Chi-Square</th>
<th>Pr &gt; ChiSq</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>-5.3095</td>
<td>1.1337</td>
<td>21.9350</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>age</td>
<td>1</td>
<td>0.1109</td>
<td>0.0241</td>
<td>21.2541</td>
<td>&lt;.0001</td>
</tr>
</tbody>
</table>

Odds Ratio Estimates

<table>
<thead>
<tr>
<th>Effect</th>
<th>Estimate</th>
<th>95% Wald Confidence Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>1.117</td>
<td>1.066 1.171</td>
</tr>
</tbody>
</table>

Classification Table

<table>
<thead>
<tr>
<th>Prob Event</th>
<th>Incorrect</th>
<th>Correct</th>
<th>Percentages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>27</td>
<td>45</td>
<td>26.2</td>
</tr>
<tr>
<td>Level 2</td>
<td>12</td>
<td>16</td>
<td>20.0</td>
</tr>
<tr>
<td>Total</td>
<td>41</td>
<td>61</td>
<td>26.2</td>
</tr>
</tbody>
</table>

Linear Logit Model Expectations

$\text{LogOdds} = \beta_0 + \beta_1 \cdot \text{(Age)}$

$\beta_0 = -5.3$  $\beta_1 = +0.11$
Logistic result on Age and CHD

Logistic Regression: Coronary Heart Disease as a function of Age
Hosmer & Lemeshow Example

3. Binary Outcomes with “Classification Trees”

CART Techniques

CART is based on the utterly simple ideas that we can deal with a multivariate prediction/classification by (a) using only parts of the database at any time (partition, segmentation), and then (b) doing some procedure over and over again in subsets (i.e., recursively).

In steps:
0. We define one DV (binary) and a set (many, p) of IVs.
1. We search for the single best predictor of the DV among ALL IVs, considering ALL “ordered splits” of IVs (i.e., all 2x2 tables).
2. We split the data (partition) into two parts according to the optimal splitting rule developed in step 1 (for 2x2 tables).
3. We reapply the search strategy on each part of the data.
4. We do this over and over again (recursively) until a final split is not warranted or a “stopping criterion” has been reached.

Clearly this would not be possible without modern day computers.

Using Salford’s SPM (CART 6.0) on CHD data from Hosmer & Lemeshow
Initial SPM 6.0 Model Selection (Pruned)

Additionally Useful Tree Details

Learning → Testing Sample Accuracy

Overall Prediction Success
Considering DTA Utility Weights (Priors)

A more elaborate Tree

Impossible Calculations?
**Complete Success?**

Initial CART 6.0 Model Selection (but this is Pruned)
Focus on the initial subset (n=100)

Could focus on the n=43 who have it!

Age Plots of (a) Logit Model Probabilities and (b) full CART Model Probabilities

4. Continuous Outcomes with “Linear” Regression
**Linear:**

\[ BD = B_0 + B_1 \times ED(\text{years}-11) \]

**Regression “Interaction” Script**

```plaintext
TITLE2 'Adding Centering and Interactions';
DATA new_data;
SET old_data;
yage50 = yearsage - 50;
yed12 = yearsed - 12;
prodaged = yage50 * yed12;
hsgrad=0; IF (yearsed GE 12) THEN hsgrad=1;
RUN;
PROC CORR NOPROB;
VAR pc_bd yage50 yed12 prodaged hsgrad;
RUN;
TITLE4 'Continuous Variable2 way ANOVA + T-Test';
PROC REG
DATA=groups2;
model1: MODEL pc_bd = yed12 /
STB
model2: MODEL pc_bd = yage50 yed12 /
STB
model3: MODEL pc_bd = yage50 yed12 prodaged / STB;
model1: MODEL pc_bd = hsgrad /
STB;
RUN;
```

**Two variable linear model results**

**Model: MODEL2**

Dependent Variable: pc_bd

Analysis of Variance

<table>
<thead>
<tr>
<th>Source</th>
<th>DF</th>
<th>Sum of Squares</th>
<th>Mean Square</th>
<th>F Value</th>
<th>Pr &gt; F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>2</td>
<td>270165</td>
<td>135083</td>
<td>379.17</td>
<td>&lt;.0001</td>
</tr>
<tr>
<td>Error</td>
<td>1877</td>
<td>668694</td>
<td>356.25695</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Corrected Total</td>
<td>1879</td>
<td>938859</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Root MSE: 18.8787

Dependent Mean: 52.85461

R-squared: 0.2878

Adj R-squared: 0.2870

Parameter Estimates

<table>
<thead>
<tr>
<th>Variable</th>
<th>DF</th>
<th>Parameter Estimate</th>
<th>Standard Error</th>
<th>t Value</th>
<th>Pr &gt;</th>
<th>Standardized Estimate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>1</td>
<td>51.25935</td>
<td>0.52158</td>
<td>98.28</td>
<td>&lt;.0001</td>
<td>0</td>
</tr>
<tr>
<td>yage50</td>
<td>1</td>
<td>-0.35293</td>
<td>0.02358</td>
<td>-14.96</td>
<td>&lt;.0001</td>
<td>-0.29687</td>
</tr>
<tr>
<td>yed12</td>
<td>1</td>
<td>2.80412</td>
<td>0.14112</td>
<td>19.87</td>
<td>&lt;.0001</td>
<td>0.39418</td>
</tr>
</tbody>
</table>

**Multiple Regression on the WAIS—R data summary**

*Part II: Multiple Regression analysis: A recommended alternative analysis*
5. Continuous Outcomes with “Regression Trees”

Regression Tree Techniques
CART is based on the utterly simple ideas that we can deal with a multivariate prediction/classification by (a) using only parts of the database at any time (partition, segmentation), and then (b) doing some procedure over and over again in subsets (i.e., recursively).

In steps:
0: We define one DV (continuous) and a set (many, p) of IVs.
1. We search for the best predictor of the DV among ALL IVs, considering ALL “ordered splits” of IVs (essentially largest t-test).
2. We split the data (partition) into two parts according to the optimal splitting rule developed in step 1 (largest t-test).
3. We reapply the search strategy on each part of the data.
4. We do this over and over again (recursively) until a final split is not warranted or a “stopping criterion” has been reached.

Clearly this would not be possible without modern day computers.
6. Classification Based on Continuous Outcomes

- Clustering or Cluster Analysis is primarily used in the situations where we have multiple outcome variables \( f(m) \) and we think we may have many independent but unobserved groups or clusters \( C \) in our data.

- The techniques of CLUSTER are empirical or exploratory and not typically used to test specific or general hypotheses about group differences (although some researchers uses these techniques that way!)

- This is related to any “search for heterogeneity” and is even used as the primary basis of “data mining techniques” (i.e., CART).
**Motivations for Clustering-Taxonomy**

- The use of CLUSTER analysis seems very well motivated mainly because there are likely to be clusters of individuals in any single group!

- There are many natural situations where we expect to see clusters, e.g., species differentiation. This is an attempt to “Carve Nature at its Joints” (Plato as reported by Meehl). Another quote from Meehl → “There are Gophers and Chipmunks, but no Gophmunks!”

- The are many situations where types, syndromes, classes and taxa are useful, e.g., clinical diagnosis and therapy. But this is used in education and marketing as well. (Meehl noted there are formal differences between Classes and Taxa – All taxa are classes, but not all classes are taxa.)

**Multivariate Clustering-Taxonomy**

- The overall benefits of the CLUSTER analysis technology comes when the results seem to make sense, and thus lead to new hypotheses to evaluate using other data.

- The overall failures of the CLUSTER technology comes when we try to use exact statistical tests of fit (there are many), and we claim that the clusters are replicable (they often are not).

- Often overlooked is the need for good selection of variables for the appropriate inferences, the scaling of these variables, and the careful interpretation of the resulting CLUSTER differences.

---

**Scatterplot of the Observed Group on Two Variables**

<table>
<thead>
<tr>
<th>Y(1)</th>
<th>All Data</th>
<th>95% Confidence Ellipsoids of our data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y(2)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Scatterplot of Two Unobserved Clusters on Two Variables**

<table>
<thead>
<tr>
<th>Y(1)</th>
<th>Cluster 1 Centroid</th>
<th>95% Confidence Ellipsoids</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y(2)</td>
<td>Cluster 2 Centroid</td>
<td></td>
</tr>
</tbody>
</table>
**Approaches to Clustering-Taxonomy**

- We define cluster membership as a person being more like one cluster than another cluster.

- More formally, unobserved heterogeneity is defined as a set of \( C \) clusters where the between cluster variability (\( B_C \)) is bigger than the within cluster variability (\( W_C \)).

- The standard solutions are typically broken down into two approaches: (1) Hierarchical and (2) Non-hierarchical. Many people use both approaches for cross-validation.

- Here we add clustering by (1) Persons or (2) Variables. Of course, we could add more dimensions to this list.

**Non-Hierarchical Cluster Approach**

- The number of clusters \( C \) is defined by the researcher.

- We assign each case to one of the \( C \) clusters based on the Euclidean distance to the cluster centroid.

- In the popular K-Means approach, we use a “sequential threshold” method for the clustering, so cases are assigned one at a time rather than simultaneously.

- Initial seeds are chosen at random or selected empirically (and this selection can make a difference).

- The final solution can be evaluated in many different ways, including the use of MANOVA in reverse (but without probability).

- To interpret the clusters we can use a MANOVA with variables that have not been used in the clustering.

---

**PROC CORR**

```
TITLE2 'Initial descriptions';
PROC CORR NOPROB;
VAR yearsage yearsed Z_vocab Z_blocks; RUN;
TITLE ''; TITLE2 '';
PROC GPLOT; PLOT Z_vocab*Z_blocks; RUN;
TITLE2 'Non-Hierarchical Clustering';
PROC FASTCLUS
MAXCLUSTERS=1 MEAN=TEMP OUT=fastout1;
VAR Z_vocab Z_blocks; RUN;
PROC FASTCLUS
MAXCLUSTERS=2 MEAN=TEMP OUT=fastout2;
VAR Z_vocab Z_blocks; RUN;
PROC PRINT DATA=fastout2; RUN;
PROC CORR DATA=fastout2;
VAR yearsage yearsed Z_vocab Z_blocks CLUSTER DISTANCE; RUN;
```

---

**WAIS-R Vocabulary and Block Design**

```
proc corr data=fastout2;
var yearsage yearsed Z_vocab Z_blocks;
run;
```
Two Clusters for Two Variables

The FASTCLUS Procedure

Replace=FULL  Radius=0  Maxclusters=2  Maxiter=1

Initial Seeds

Cluster           Z_Vocab          Z_Blocks

1         -2.727525054      -2.319485247
2          1.603005388       2.126454738

Criterion Based on Final Seeds =   0.7005

Cluster Summary

Maximum Distance

<table>
<thead>
<tr>
<th>Cluster</th>
<th>Frequency</th>
<th>RMS Std</th>
<th>Deviation</th>
<th>from Seed</th>
<th>Radius</th>
<th>Nearest</th>
<th>Distance Between</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>551</td>
<td>0.7629</td>
<td>2.7820</td>
<td>2</td>
<td>2.1494</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1129</td>
<td>0.6682</td>
<td>2.2602</td>
<td>1</td>
<td>2.1494</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Statistics for Variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Total STD</th>
<th>Within STD</th>
<th>R-Square</th>
<th>RSQ/(1-RSQ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Z_Vocab</td>
<td>1.00000</td>
<td>0.68594</td>
<td>0.529767</td>
<td>1.126607</td>
</tr>
<tr>
<td>Z_Blocks</td>
<td>1.00000</td>
<td>0.71499</td>
<td>0.489096</td>
<td>0.957316</td>
</tr>
</tbody>
</table>

OVER-ALL | 1.00000      | 0.70061         | 0.509432 | 1.038453    |

Pseudo F Statistic =  1742.52

Approximate Expected Over-All R-Squared =   0.37575

Cubic Clustering Criterion =   22.536

WARNING: The two values above are invalid for correlated variables.

Cluster Means

Cluster           Z_Vocab          Z_Blocks

1         -1.041560545      -1.000780939
2          0.508326212       0.488424017

Clusters of Vocabulary and Block Design

Further Evaluation by MANOVA

TITLE2 'Creating Biased GLM';
PROC GLM DATA=fastout2;
CLASS CLUSTER;
MODEL Z_Vocab Z_Blocks = CLUSTER;
MANOVA H=CLUSTER / CANONICAL;
RUN;

TITLE2 'Creating UN-Biased GLM';
PROC GLM DATA=fastout2;
CLASS CLUSTER;
MODEL YearsAge YearsEd = CLUSTER;
MANOVA H=CLUSTER / CANONICAL;
RUN;
**Pseudo MANOVA for Checking**

**Real MANOVA for Interpretation**

Computer Programs for Mixture Models

1. Mplus (Muthen & Muthen, 2008) is very flexible and popular, but is somewhat expensive (but check out R).

2. Latent GOLD (or LEM) by Vermunt & Magidson (2009) is expensive but excellent (but can get LEM for free).

3. A lot of free R-code is available – LCA (Wallwe, 2004) and MCLUS (Rafferty, 2009) OpenMx (boker et al, 2013) and all seem to work well.

4. GLAMM (Skrodal & Rabe-Hesketh, 2004) is a STATA add on.

5. BMDP LE (see Pearson et al, 1992) is still used by many.
Comparison of Two-group Cluster & Mixture Models of WAISR *

<table>
<thead>
<tr>
<th>Cluster 1</th>
<th>Cluster 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>244 (14.5%)</td>
<td>307 (18.3%)</td>
</tr>
<tr>
<td>12 (0.01%)</td>
<td>1,117 (66.5%)</td>
</tr>
</tbody>
</table>

Agreement vs. Disagreement in Class Probabilities

<table>
<thead>
<tr>
<th>Agreements</th>
<th>Probability in Class 1</th>
<th>Probability in Class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>N 244</td>
<td>Average 0.908</td>
<td>Minimum 0.507</td>
</tr>
<tr>
<td>N 1117</td>
<td>Average 0.995</td>
<td>Minimum 0.564</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Disagreement</th>
<th>Probability in Class 1</th>
<th>Probability in Class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>N 12</td>
<td>Average 0.704</td>
<td>Minimum 0.505</td>
</tr>
<tr>
<td>N 307</td>
<td>Average 0.946</td>
<td>Minimum 0.521</td>
</tr>
</tbody>
</table>

The Practical Need for Mixture models

Several key questions about data analysis are very hard to deny → “Heterogeneity is the rule not the exception…” (Cattell, Coulter & Tatsujioka, 1966)
“A taxon is a type, a natural kind…All taxa are classes, but not all classes are taxa…There are natural groupings of people, just as in the animal kingdom – there are Gophers and Chipmunks, but there are no Gophmunks! (Meehl, 1992, p.333)
“This approach, which combines aspects of finite mixture models and GLMs, provides a comparatively fast learning and good generalization for nonlinear regression problems, including classification. It is thus a serious competitor to well-known statistical procedures such as MARS and CART.” (McLachlan & Peel, 2000, p.4)
8. Introducing SEM+Trees
In the behavioral and social sciences, structural equation models (SEMs) have become widely accepted as a modeling tool for the relation between theoretical or “latent” variables and observed or “manifest” variables.

SEMs can be seen as a unification of several multivariate analysis techniques, as long as there is an initial model.

What we generally do is define some kind of SEM over all people and fit it using a SEM program.

Misfits are used to determine where the individual’s membership in a group should be.

In the behavioral and social sciences, decision tree analysis (DTAs or TREES or CART or SEARCH) has become widely accepted as a modeling tool for the relation between one observed “outcome” variable and multiple “predictor” variables.

TREES can be seen as a unification of several multivariable prediction techniques, as long as there is a target model.

What we generally do is define some kind of DTA over all variables and fit it using an search and recursive program.

Misfits are used to determine where the group membership is known or unknown.

In essence, a multiple group model with all parameters invariant in the SEM is fit to splits of ALL the “observed variables not in the target model”, and we look for the split with the biggest misfit. The following splits are all done “recursively” (following Sonquist & Morgan, 1963, 1970).

In this sense, SEM-Trees offers the first true “multivariate” TREE analysis (no it is not MARS!), and it is “new.”

In this presentation we describe the new methodology, discuss theoretical and practical implication, and we demonstrate applications of various models of interest.
**How do we calculate any DTA?**

The key idea is that we have a measured series and we fit a mixed model to a groups of participants, then we end up with a scalar from an individual likelihood:

\[ L^2_n\{\mu, \Sigma\} = K + (Y_n - \mu) \Sigma^{-1} (Y_n - \mu)' \]

where the individual’s scores (Y) are compared to a population mean (\(\mu\)) and population covariance (\(\Sigma\)) defined by a model (and \(K\) is a constant).

The total misfit for any model is the sum of these likelihoods, written as:

\[ \text{Overall } L^2\{\mu, \Sigma\} = \sum L^2_n\{\mu, \Sigma\} \]

Isolating each person’s misfit can be a useful way to examine influential observations (see McArdle, 1997)

---

**Splits of the Data by Measured Variables**

Now we can split the people into groups defined by the measured splitting variable (X) into subsets a and b, and form a new likelihood for each person as:

- If \(X=a\), \( L^2_{na}\{\mu_a, \Sigma_a\} = K + (Y_{na} - \mu_a) \Sigma_a^{-1} (Y_{na} - \mu_a)' \)
- If \(X=b\), \( L^2_{nb}\{\mu_b, \Sigma_b\} = K + (Y_{nb} - \mu_b) \Sigma_b^{-1} (Y_{nb} - \mu_b)' \)

- The total misfit for this joint model is the weighted sum of these likelihoods, written as
  \[ \text{Split } L^2\{\mu, \Sigma\} = \sum L^2_{na}\{\mu_a, \Sigma_a\} + \sum L^2_{nb}\{\mu_b, \Sigma_b\} \]
- So the effectiveness of the split of data is defined by the likelihood difference – if it is large they should be split!
- \( \text{Gain } L^2\{\mu, \Sigma\} = \text{Split } L^2\{\mu, \Sigma\} - \text{Overall } L^2\{\mu, \Sigma\} \)

This can be done automatically as long as split variables are “measured.” This is not a latent class problem.

---

**Standard R versions of Reg and CART**

```r
> model.2
> x1.2 = (x1 > 0) # note the use of a cutoff at 0,
> so it only varies after 0
> x2.2 = (x2 > 0) # as above
> y.2 = 1 + (2*x1.2) + (3*x2.2) + ey
#Linear regression
> reg.2 <- lm(y.2~x1.2+x2.2)
> summary(reg.2)

Residuals:
Min       1Q   Median       3Q      Max
-3.07219 -0.68178  0.00666  0.72954  3.09040

Coefficients: Estimate Std. Error t value  Pr(>|t|)
(Intercept)     1.02631    0.05499   18.66   <2e-16 ***
x1.2TRUE     2.00439    0.06451   31.07   <2e-16 ***
x2.2TRUE     2.91782    0.06449   45.25   <2e-16 ***
Residual standard error: 1.02 on 998 degrees of freedom
Multiple R-squared: 0.7527, Adjusted R-squared: 0.7522
F-statistic:  1519 on 2 and 998 DF,  p-value: < 2.2e-16
```

---

**R Regression Output**

```
Call:
  lm(formula = y ~ x1 + x2)
Residuals:
   Min     1Q   Median     3Q    Max
-3.0847 -0.6854  0.0149  0.7092  3.9269
Coefficients: Estimate Std. Error t value  Pr(>|t|)
(Intercept)  1.02756    0.03310   31.04   <2e-16 ***
x1          1.90252    0.03197   59.52   <2e-16 ***
x2          2.99095    0.03332   89.77   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.045 on 998 degrees of freedom
Multiple R-squared: 0.919, Adjusted R-squared: 0.919
F-statistic:  5673 on 2 and 998 DF,  p-value: < 2.2e-16
```
**R RPART Output**

```r
call: rpart(formula = y.2 ~ x1.2 + x2.2, method = "anova")
n= 1001
CP nsplit rel error xerror xstd
1 0.5134325 0 1.0000000 1.0019140 0.03580360
2 0.1295204 1 0.4865675 0.4890493 0.01906300
3 0.1100367 2 0.3570472 0.3585805 0.01558064
4 0.0100000 3 0.2470104 0.2486406 0.01109804
```

Node number 1: 1001 observations, complexity param=0.5134325
mean=3.444344, MSE=4.194623
left son=2 (506 obs) right son=3 (495 obs)
Primary splits:
- x2.2 < 0.5 to the left, improve=0.5134325, (0 missing)
- x1.2 < 0.5 to the left, improve=0.2453740, (0 missing)

Node number 2: 506 observations, complexity param=0.1295204
mean=1.992849, MSE=2.147889
left son=4 (262 obs) right son=5 (244 obs)
Primary splits:
- x1.2 < 0.5 to the left, improve=0.5003832, (0 missing)

Node number 3: 495 observations, complexity param=0.1100367
mean=4.928094, MSE=1.93167
left son=6 (252 obs) right son=7 (243 obs)
Primary splits:
- x1.2 < 0.5 to the left, improve=0.4831997, (0 missing)

Node number 4: 262 observations
mean=0.9923852, MSE=0.9943195

Node number 5: 244 observations
mean=3.067118, MSE=1.157737

Node number 6: 252 observations
mean=3.979386, MSE=1.064824

Node number 7: 243 observations
mean=5.91194, MSE=0.9292872

---

**The OpenMx RAM notation requires every path in the diagram to be defined.**

![RAM Notation Diagram]
**RAM specification in OpenMX**

```r
REG2_Model <- mxModel("Multiple Regression Model Path Specification", type="RAM", manifestVars=c("y", "x1", "x2"), latentVar="ey").
# first direct impacts
mxPath(from="x1",to="y", arrows=1, free=TRUE, value = .1, label = "B1"),
mxPath(from="x2",to="y", arrows=1, free=TRUE, value = .1, label = "B2"),
# adding residual path
mxPath(from="ey",to="y", arrows=1, free=FALSE, values = 1, label="fixed 1"),
# model variances and covariances
mxPath(from="x1", arrows=2, free=TRUE, value = 1, label="V1"),
mxPath(from="x2", arrows=2, free=TRUE, value = 1, label="V2"),
mxPath(from="ey", arrows=2, free=TRUE, value = 1, label="Ve"),
mxPath(from="x1", to= "x2", arrows=2, free=TRUE, values = .1, label="C12"),
#add required means in column notation
mxPath(from="one",to=c("y", "x1", "x2"),arrows=1, free=TRUE, values = c(1,1,1), labels=c("B0", "Mx1", "Mx2") )
) # close model
```

After this, adding a SEMTree to any SEM is easy

```r
# build a tree
treel <- semtree(REG1_Model)
# plot the result
plot(treel)
```

**Results of OpenMX**

```r
free parameters:
name matrix row col Estimate Std.Error lbound ubound
1 B1 A y x1 1.9941558799 0.03077067
2 B2 A y x2 3.0290359794 0.03212727
3 V1 S x1 x1 1.0968105126 0.04902384
4 C12 S x1 x2 -0.0009623423 0.03320199
5 V2 S x2 x2 1.0061455782 0.04496839
6 Ve S ey ey 1.0395564636 0.04646684
7 B0 H 1 y 1.0055534064 0.03223348
8 Mx1 M 1 x1 0.0218588355 0.03309820
9 Mx2 M 1 x2 -0.0063300329 0.03170186

observed statistics: 3003
estimated parameters: 9
degrees of freedom: 2994
-2 log likelihood: 8659.608
saturated -2 log likelihood: NA
number of observations: 1001
chi-square: NA
p: NA

Information Criteria:

<table>
<thead>
<tr>
<th>df</th>
<th>Penalty Parameters</th>
<th>Penalty Sample-Size Adjusted</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIC</td>
<td>2671.608</td>
<td>8677.608</td>
</tr>
<tr>
<td>BIC</td>
<td>-12025.203</td>
<td>8721.787</td>
</tr>
<tr>
<td>CFI</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>TLI</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>RMSEA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>
```

Figure 2: Illustration of a structural equation model (SEM) Tree with a linear growth curve model as the
underlying model. Black nodes represent exogenous variables. Red nodes indicate the SEM Tree.

28
**How do we calculate a Multivariate DTA?**

The key idea is that we have a measured series including multiple outcome variables (e.g., a factor analysis model, a latent change score model) and we fit a two group model to two groups of participants, then we end up with the scalar from of an individual likelihood $L^2_{n}(\mu, \Sigma)$.

We can calculate the total misfit for any model as the sum of these likelihoods. Isolating each person’s misfit can be a useful way to examine influential observations (see McArdle, 1997).

But now the overall model can be fitted to each sub-group that may or may not fit the pattern, and we will know from the invariance likelihood that they are not the same.

**SEM-Trees model limitations**

SEM-Trees is relatively new, so it is not clear if “optimal” and “repeatable” splits are found using the available software (i.e., semtrees).

SEM-Trees focuses attention on all measured variables so all key variables are assumed to be measured – this may be a bit different than Latent Growth/Mixture models, but it is the same as any model where we assume the class is defined by a measured variable.

SEM-Tree are clearly exploratory, so $p$-values are hard to use effectively, and permutation tests and bootstrap tests may be useful. Of course, it is easy to forget this and pretend to do a model with best contrasts!

---

**9. Summary & Discussion**

**Alternative DTA/CART “Data Mining”**

- One common feature of these new techniques is that they are “exploratory” and rely on “computer assisted” analysis.
- Three kinds of DTA/CART based on the outcome – (1) Binary Classification, (2) Regression Classification, and (3) Unsupervised Supervision.
- The big sub-division of these techniques is where we uses a single outcome and tries to make a optimal prediction of this outcome from multiple IVs– these are generally termed “supervised learning” techniques (including DTA/CART).
- Another subdivision does not require an outcome and merely classifies persons into subgroups based on similarities among a set of variables – not surprisingly these are termed “unsupervised learning” techniques.
When should we use DTA/CART?

**DO NOT use DTA/CART when:**
- You are beginning, and developing hypotheses.
- You are at the confirmatory/testing stage of research.

**DO use DTA/CART when:**
- You have defined some key outcomes (DV).
- You are at the exploration stage of research.
- You believe there are higher order interactions/nonlinearity in the correct model.
- You want to calculate an “optimal prediction model” -- possibly as a more reasonable basis for evaluating your other models.

TODAY!

### DTA/CART is Easy and Hard

**The EASY Parts:**
- DTA/CART is an automated stepwise procedure that considers most nonlinearity and all interactions.
- DTA/CART does not need assistance with scaling and scoring and variable redundancy is allowed (we put all IVs in).
- DTA/CART does a lot of numerical searching, but it is very fast.

**The Hard Parts:**
- What is the key outcome variable?
- How many nodes should be used?
- Is the sample of persons a goo representation of the data?
- DTA/CART handles missing data, but in many ways.

---

**Key References for DTA/CART Overviews**

**Newest SEM-Trees References**