DEPARTMENT OF MANAGEMENT

AFDELING FOR VIRKSOMHEDSLEDELSE

Working Paper 2006-7

Analyzing Customer Lifetime Value using Tree Ensembles

Niels Stender

UNIVERSITY OF AARHUS • DENMARK

ISSN 1398-6228

Analyzing Customer Lifetime Value using Tree Ensembles

Niels Stender
University of Aarhus
niels@innohead.com

June 9, 2006

Abstract

This paper presents a quantitative method for gaining knowledge about patterns in customer behavior and related business revenue generation, based on customer transaction histories and a customer relationship statespace. The method enables the study of patterns conditional on a large set of individual customer characteristics, requiring minimal assumptions about the functional form of this relation.

Assuming that customer behavior follows a markov decision process and given a discount factor and revenue generated in each relationship state, the internal variance of the life-time value of a group of customers can be calculated. Given a set of customer characteristics, an ensemble of decision trees is grown by searching for customer characteristics that will split the sample into two groups that minimize internal variance of the life-time value. The trees are pertubed by bagging and random feature selection, thus enabling out of sample fit measures to be calculated, along with measures of variable importance and observation similarity.

The model is applied in a study of a newspaper's customer database, where event histories are linked to sociodemographic variables.

Contents

1 Introduction

Discovering patterns in customer behavior and revenue generation is a very helpful ingredient in business decision making. The marketing and management community have in recent years introduced concepts such as customer relationship management, customer equity and customer life-time value to address the appetite for frameworks to understand customer-revenue dynamics. To apply these concepts to real data, a range of quantitative methods have been utilized, such as event history analysis and duration or hazard-models.

In my own study of various companies' customer transaction databases, I felt a better method-toolbox was needed. Traditional methods have difficulties meeting the challenges of the typical scenario faced by a business analyst. The typical scenario involves:

- Many customer transaction histories
- Many explanatory variables
- Little willingness to assume anything about the relation between customer transaction histories and explanatory variables

Parametric methods such as linear regression, logit/probit models and basic hazard models involve rather restrictive assumptions about the functional relation of transaction histories and explanatory variables, a critique that applies equally well to a range of semi-parametric models that are flexible with respect to unobserved heterogeneity, such as the Cox-model. Methods from machine learning, such as decision trees and neural networks, and non-parametric statistical methods such as conditional kernel regression, address the problem of specification in increasing degree of generality. Decision trees are very easy to understand and apply, but their predictive ability is dominated by neural networks and conditional kernel density regression in most realistic cases. Neural networks are difficult to set up and computationally expensive, while conditional kernel density regression and the search for an optimal bandwidth-vector does not scale well in the number of explanatory variables with respect to algorithm time consumption.

Ensembles of decision trees or random forests, as described in Breiman (2001), seem to improve on many of these limitations. They are flexible, can handle many explanatory variables, seem robust in noisy datasets, see Hamza & Larocque (2005), and scale well with respect to time consumption.

In marketing papers there have been few applications of tree ensembles. One is Buckinx & Van den Poel (2005), another is Lariviere & Poel (2005). Common for these and many other quantitative models of customer lifetime value is the lack of a unified theoretical approach. Tree ensembles and other flexible methods have been used to model and predict a discrete or continous dependent variable outcome. This paper develops a variant of tree ensembles that apply to full customer event histories on a markov chain, sometimes referred to as a markov decision process, and a derived lifetime value score.

The main source and inspiration for the tree ensemble implementation has been Breiman (2001), Breiman & Cutler (2001)'s fortran source code and the R-implementation authored by Andy Liaw and Maathew Wiener, described in Liaw & Wiener (2002).

2 Customer Behavior on a Markov Chain

What is customer behavior? Any person that might already be a customer, a former or future customer, is sometimes referred to as an agent. The status of the relation between an agent and a business is thought to be in one distinct state at any point in time. The set of all possible states is the relationship statespace. Customer behavior is transitions between states.

A newspaper subscription database is studied in a subsequent section, so let an imaginary newspaper serve as an illustration of these ideas. In one scenario we might decide to classify agents from year to year as

- 0: Not a customer
- R: Rebate/discount program. The customer participates in an introductory subscription program, to allow him to sample the newspaper.
- F: Full price. The customer has entered a full price subscription program.

The relation statespace contains three states {0, R, F}. A typical event history could then follow patterns like, say, {R,0}, {R, F, 0} and {R,F,F,0,0,R,F}. The first event history represents a customer who accepted a discount program, but didn't enter into full price subscription. The second event history represents a customer who also entered a discount program, then accepted entry into full price subscription, but churned after one period. The final event history also represents someone entering a discount program, follow by two periods of full-price subscription. He then churns, but acceptes entry into a discount program a second time.

Are there any regularities in customer behavior? We will assume that customers live on a discrete Markov process, the Markov chain.

Assume we are given a relationship statespace $\mathcal{S} = \{0, ..., S\}$. An event history is a sequence of states $h = \{s_i\}, \forall i : s_i \in \mathcal{S}$. The transition probability function p describes the probability of observing a state conditional on various information. In particular, if the probability of observing a given state is a function of nothing but the prior state occupied, then the system is said to have the Markov property and can be said to be a Markov chain. A Markov transition matrix is a matrix of $|\mathcal{S}| \times |\mathcal{S}|$ elements, such that element (i, j) contains $p(s_{t+1} = j | s_t = i)$. A Markov transition matrix completely describes a Markov chain.

First we will assume that all agents live on the same chain, while later introducing separate chains for different groups of customers, that is, a transition probability function that vary with agent characteristics.

$$\begin{bmatrix} & 0 & R & F \\ 0 & 0.99 & 0.01 & 0 \\ R & 0.8 & 0 & 0.2 \\ F & 0.1 & 0 & 0.9 \end{bmatrix} \tag{1}$$

Given the {0, R, F} statespace and a transition matrix as in eq. 1, we would say that the probability of moving from "0" to "R" in a given year is 0.01. To go on from "R" to "F" is related with a probability of 0.2. A transition from "F" and back to "0" would happen with probability 0.1.

The imagined case serves to illustrate some obvious problems with the Markov property. Think of event history number three. A former full price-paying customer in state "0" is just as likely as a newbie to enter into a rebate program. It would be more reasonable to think that the former customer is either much less likely, because of dissatisfaction, or much more likely to come back, since he has displayed some interest in the product earlier on. A second example is that of a very loyal customer, who has been subscribing to the full price package for many years. He is as likely to churn as someone who just entered that state.

One solution to that particular problem is to expand the statespace. One could insert states some variant "0"-states, say, "0F" and "0R", for agents having churned from the full price or the rebate program. To model the loyalty aspect, variant "F"-states, say, "F.1", "F.2" and so on, could be inserted. The first year full price subscribers would enter "F.1". Those surviving from "F.1" could enter "F.2" and so on.

Expanding the statespace is not free however, as we are introducing more parameters into the model. Balancing model complexity against the available number of observations and computier power is a problem facing any empirical researcher.

2.1 Customer Value

From this point and onwards it is assumed that the period-value of an agent's state-occupany can be specified. Let the the revenue vector r contain this information.

In subsequent sections we want to group agents together according to similarity in behavior. There are many methods for judging similarity, such as likelihood or the gini-measure, but for a business the notion of life-time value is an appropriate measure. It is more important to predict the overall discounted value of a future event history right then it is to estimate probabilities in the transition matrix accurately. The two goals are related, but the issue lies in weighting the errors when minimizing the overall error, say, with respect to the tradeoff between the (0,0) transition versus the (F,F) transition. Given an abundance of 0-state agents, and few F-state agents, a likelihood approach would result in a heavy weighing of errors in the 0-state. With respect to the revenue stream flowing to the company, we would want to weight the F-state relatively more.

What is the present value of an agent? Given a discount factor β, if we knew the future event history with certainty, the lifetime value v would be as stated in eq. 2. Let r_i denote the i'th element in vector r.

$$v = r_{s_0} + \beta r_{s_1} + \beta^2 r_{s_2} + \ldots = \sum_{i=0}^{\infty} \beta^i r_{s_i} \tag{2}$$

In the face of uncertainty we are looking for the expected value of a discounted stochastic revenue flow. Let M be a transition probability matrix for an agent, for example, as in eq. 1. Looking one period into the future, we can expect to earn $v_0 = Mr$, the expected revenue for state i shown in the i'th element of v_0. In the course of two periods, we can expect to earn, in discounted value, $v_1 = Mr + \beta M v_0$. So,

$$
\begin{aligned}
v_0 &= Mr \\
v_1 &= Mr + \beta M v_0 = Mr + \beta MMr \\
v_2 &= Mr + (\beta M)\, Mr + (\beta M)^2\, Mr \\
&\vdots \\
v_n &= \left(I + \beta M + (\beta M)^2 + \ldots + (\beta M)^{n-1}\right) Mr + (\beta M)^n\, Mr,
\end{aligned}
$$

I being the identity matrix. Taking v_n to the limit and observing that no entry in M is larger than one and assuming that $\beta \in [0, 1[$ it will be true that each element of $\lim_{n\to\infty} (\beta^n M^n)$ is zero. Thus we are left with a geometric series in βM, leading to the useful representation in eq. 3.

$$E(v) = \lim_{n\to\infty} v_n = (I - \beta M)^{-1}\, Mr \tag{3}$$

Using the {0,R,F} statespace as an example, let $\beta = 0.9$ per year. Assume our revenue from non-customers is \$0, the discount program costs \$10/year and full price customers earns us \$20/year, so $r = \begin{pmatrix} 0 & -10 & 20 \end{pmatrix}'$.

$$
\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - 0.9 \begin{bmatrix} 0.99 & 0.01 & 0 \\ 0.8 & 0 & 0.2 \\ 0.1 & 0 & 0.9 \end{bmatrix} \right)^{-1} \begin{bmatrix} 0.99 & 0.01 & 0 \\ 0.8 & 0 & 0.2 \\ 0.1 & 0 & 0.9 \end{bmatrix} \begin{bmatrix} 0 \\ -10 \\ 20 \end{bmatrix} = \begin{bmatrix} 0.879\,33 \\ 21.761 \\ 95.153 \end{bmatrix}
$$

Agents currently in state 0, R and F is worth, respectively, \$0.88, \$21.8 and \$95.1 in present value dollars.

Turning our attention to that of the variance of v, we observe that one period ahead, the expected squared deviation will be $u_0 = M(r \circ r) - (Mr) \circ (Mr)$. Sobel (1982) demonstrates how to derive the variance of a revenue-markov chain, the result shown in eq. 4. The measure in eq. 4 is also called internal variance.

$$
\begin{aligned}
var(v) &= \left(I - \beta^2 M\right)^{-1} \theta, \\
\theta &= M(r + \beta v) \circ (r + \beta v) - v \circ v
\end{aligned} \tag{4}
$$

$\circ$ Hadamard product/element-by-element multiplication.

Crunching the numbers for the {0,R,F } case results in

$$var(v) = \begin{pmatrix} 126.11 & 819.08 & 3325.7 \end{pmatrix}^T,$$

or element-by-element standard deviation

$$sd(v) = \begin{matrix} 11.230 & 28.620 & 57.669 \end{matrix}.$$

3 Tree Ensembles

This section deals with the challenge of searching for relations between customer lifetime value and customer attributes. As mentioned in the introduction, tree ensembles seem to have some advantages when many explanatory variables are present, when the functional form of the relation betweeen dependent and independents are unknown and when computing power is limited.

A tree ensemble is a special case of a classifier ensemble, in that the classifiers are decision trees. In this paper, each tree is built using random feature selection at each split combined with bagging. When predicting a dependent variable from an attribute vector, the estimate is formed by some average of the predictions from individual trees in the ensemble. Breiman (2001) refers to such a type of ensemble as a random forest.

A decision tree is a hierarchical partition of data, usually done to optimize some measure of similarity. The most common decision tree is the binary tree, that iteratively splits data into two groups by searching for a single attribute, and a threshold value, that minimizes weighted dissimilarity within the two partitions formed by splitting data on the attribute: One group for observations with an attribute value below the thresold value, another group for observations with an attribute value above or equal to the thresold value. The iteration is terminated when the number of observations in a leaf drops below a preset thresold. A function of the observations in the leaf is then used to form a prediction for all observations fitting the partition criteria.

3.1 Algorithm

Let $D = \{0, ..., n_D - 1\}$ be an index of n_D agents, and the matrix Y a collection of attributes related to each observation, element (i, j) equal to agent i's j'th attribute y_{ij}. Let $K = \{0, ..., n_K - 1\}$ be an index of the n_K attributes. The vector $h_i = \{s_{ij}\}_{j=0}^{n_i-1}$, $s_{ij} \in \mathcal{S}$ contains, in sequence, the event history of length n_i of agent i.

For a group $\hat{D} \in D$ of agents, the estimated markov transition matrix $\hat{M}_{\hat{D}}$ can be estimated as in eq. 5.

$$\hat{M}_{\hat{D}} = \left\{ \frac{\sum_{i\in\hat{D}} \sum_{j=1}^{n_i-1} 1\left(s_{i,j-1} = k \text{ and } s_{i,j} = l\right)}{\sum_{i\in\hat{D}} \sum_{j=1}^{n_i-1} 1\left(s_{i,j-1} = k\right)} \right\}_{(k,l)\in\mathcal{S}\times\mathcal{S}} \tag{5}$$

Given a markov transition matrix M, revenue vector r and discount factor β, define v_M as a stochastic vector of lifetime values. The expected value of v_M is displayed in eq. 3.

Given a group of agents $\hat{D} \in D$ and their attribute values $Y_{\hat{D}} = \{y_{ij}\}, i \in \hat{D}, j \in K$, we can define two partitioning functions f_{left} and f_{right} that forms disjoint groups by partitioning on an attribute $k \in K$ and a thresold value $y \in \{y_{ik} | i \in \hat{D}\}$. Let

$$f_{left}\left(\hat{D}, k, y\right) = \left\{i \in \hat{D} | y_{ik} < y\right\}$$
$$f_{right}\left(\hat{D}, k, y\right) = \left\{i \in \hat{D} | y_{ik} \geq y\right\}.$$

$\hat{D}$ can be associated with a vector of estimated lifetime value variance, $var\left(v_{\hat{M}_{\hat{D}}}\right)$, through equations 4 and 5. The goal is to partition $\hat{D}$ into two groups with a small overall variance. For an optimization scheme to work, we need a way of collapsing two vectors of lifetime value variance into a single number.

In the application section following later, the principles laid out in this section are used to understand a possible future flow of revenue from new customers. Therefore the estimated distribution of initial states is used to weight $var\left(v_{\hat{M}_{\hat{D}}}\right)$. In other studies, such as a study with the goal of segmenting the current customer base with respect to future value, it would be reasonable to weight by the last state in each customer's event history.

A vector of estimated initial state probabilities $f_{init}(\cdot)$ for a group of agents $\hat{D}$ is given in eq. 6.

$$f_{init}\left(\hat{D}\right) = \left(\frac{\sum_{i\in\hat{D}} 1\left(s_{i,0} = k\right)}{|\hat{D}|}\right)_{k\in\mathcal{S}} \tag{6}$$

The weighted lifetime value variance $w(\cdot)$ is as given in eq. 7.

$$w\left(\hat{D}\right) = f_{init}\left(\hat{D}\right)' var\left(v_{\hat{M}_{\hat{D}}}\right) \tag{7}$$

When partitioning $\hat{D}$, we can weight each group's lifetime value variance by the number of agents in the group, thus arriving at an overall measure for lifetime value variance as seen in eq. 8.

$$\begin{aligned} w\left(\hat{D}, k, y\right) &= \frac{|D_{left}|w(D_{left}) + |D_{right}|w\left(D_{right}\right)}{|D_{left}| + |D_{right}|} \\ D_{left} &:= f_{left}\left(\hat{D}, k, y\right) \\ D_{right} &:= f_{right}\left(\hat{D}, k, y\right) \end{aligned} \tag{8}$$

Now to the process of generating a tree ensemble. A **bagging procedure** is used, to enable the study of out-of-bag diagnostics and to prevent the model from overfitting. Bagging, bootstrapping, cross-validation and hold-out samples are all terms with similar meaning. Bagging involves the generation of a subsample of data, in-the-bag data, on which a full model or model element is estimated. The remaining data, the out-of-bag, is then used for validation purposes, such as calculating an out-of-bag fit measure.

Set m_{sample} to the number of observations to sample from D in the bagging procedure. A typical value for m_{samle} is in the range of 50% to 70% of the number of observations and only extreme values seem to alter the resulting fit.

Random feature selection. When generating the typical single decision tree, each split is formed by searching through all possible attributes. This is likely to yield a reasonably good fit, but has some problems similar to those involved in solving optimization tasks in the plane by greedy hill-climbing. Instead of growing trees using a greedy search, random feature selection fall in the class of stochastic search techniques. In each split, a random subset of the attributes are searched over.

Let m_{try} denote the number of attributes to sample from K at each split iteration. Continuing an analog to other search algorithms, this parameter can be compared to the temperature parameter in simulated annealing. A very low m_{try} can lead to slow or no progress in the fitting measure when increasing the number of trees, while setting m_{try} too high will leave too much of the search space unexplored and generate trees that look similar.

Tree split iteration is terminated when the number of in-the-bag observations in a leaf drop below m_{leaf}. The two model parameters, m_{leaf} and m_{try}, are the most important in the search algorithm and can be thought of as the tree ensemble equivivalent to the bandwidth parameter in kernel density estimation or the nearest-neighbour fraction in nearest-neighbour clustering.

Let m_{tree} equal the number of trees in the ensemble and collect in-bag and out-of-bag memberships in the $n_D \times m_{tree}$ matrix B. Element (i, j) in B is set to one if observation i is in-bag in tree j and zero otherwise.

All out-of-bag lifetime value estimates $\hat{v}_{ij}$ for agent i from tree j are collected in the $n_D \times m_{tree}$ matrix $\hat{V}$.

A tree ensemble is created by calling ensemble() as described below.

ensemble():
    for each $i \in \{0, ..., m_{tree}\}$:
        tree_root(i)

tree_root(t):
    D_{in} = sample(D, m_{sample})
    for each $i \in D_{in}$:
        B_{it} = 1
    D_{oob} = $D \backslash D_{in}$
    tree_branch(t, D_{in}, D_{oob})

tree_branch(t, D_{in}, D_{oob}):
    if $|D_{in}| \leq m_{leaf}$:
        tree_leaf(t, D_{in}, D_{oob})
    else:
        $\hat{K}$ = sample(K, m_{try})
        $(k^*, y^*) = \arg\min_{k \in \hat{K}, y \in \{y_{ik} | i \in D_{in}\}} w(D_{in}, k, y)$
        tree_branch(t, $f_{left}(D_{in}, k^*, y^*)$, $f_{left}(D_{oob}, k^*, y^*)$)
        tree_branch(t, $f_{right}(D_{in}, k^*, y^*)$, $f_{right}(D_{oob}, k^*, y^*)$)

tree_leaf(t, D_{in}, D_{oob}):
    for each $i \in D_{oob}$:
        $\hat{v}_{it} = E\left(v_{\hat{M}_{D_{in}}, s_{i0}}\right)$

sample(D, n):
    return (n elements picked from D with equal probability)

The overall lifetime value prediction for each customer can now be calculated as in eq. 9.

$$\hat{v}_i = \frac{1}{|\{j | B_{ij} = 1\}|} \sum_{\{j | B_{ij} = 1\}} \hat{V}_{ij} \tag{9}$$

3.1.1 Out-of-bag Fit Measure

When evaluating tree ensemble performance Breiman (2001) uses a mean-square error criteria for continous outcome variables, while for categorical outcomes he counts the number of correct predictions. In contrast, this paper is concerned with an empirically unobservable construct, the lifetime value of a customer.

The approach taken is to rely on local out-of-bag measures, local in the sense that it only requires information available within a tree. When creating a leaf, the expected lifetime value estimate is calculated and compared to the same calculation done on the out-of-bag observations, to get an estimate of how wrong the guess on mean estimated value might be. An out-of-bag mean square error $\hat{\sigma}^2_{oob.mean,t,i}$ is calculated in each tree t, for each out-of-bag observation i using the procedure in eq. 10.

$$\hat{\sigma}^2_{oob.mean,t,i} = \left(E\left(v_{\hat{M}_{D_{in}},s_{i0}}\right) - E\left(v_{\hat{M}_{D_{out}},s_{i0}}\right)\right)^2 \tag{10}$$

Define the tree mean-square error to $\hat{\sigma}^2_{oob.mean,t} = \hat{E}_{i\in D_{oob}}\left(\hat{\sigma}^2_{oob.mean,t,i}\right)$ and the ensemble mean-square error as $\hat{\sigma}^2_{oob.mean} = \hat{E}_{t\in\{0,\ldots,m_{tree}-1\}}\left(\hat{\sigma}^2_{oob.mean,t}\right)$. The latter is an estimate of the average error in a tree. The estimate is guaranteed to converge to some limit, since each generated tree is drawn from the same distribution with respect to how the bags are formed, how the split candidates are selected at each turn and the maximum node size. It is a different matter to interpret what inherent meaning the limit carries with it. It is not a measure of the expected error when estimating the lifetime value of a single agent, rather it is an estimate of the error in the estimated mean for a group of agents. The agents are grouped mainly as functions of m_{try} and m_{leaf}, so $\varepsilon^2_{oob.mean}$ can be thought of as a measure of the error with respect to the segmentation structure laid out by m_{try} and m_{leaf}.For example, for larger m_{leaf}, the more coarse the tree partitioning will be, that is, fewer leaves will be generated in each tree. Fewer leaves means more observations in each group, which should help in guessing the lifetime value.

A second local measure is the average internal variance of lifetime value $\sigma^2_{oob.iv.}$. For each out-of-bag observation i in each tree t, calculate the internal variance $\hat{\sigma}^2_{oob.iv,i,t} = var\left(v_{\hat{M}_{D_{oob}},s_{i0}}\right)$, based on eq. 4. Then $\hat{\sigma}^2_{oob.iv} = \hat{E}_{t\in\{0,\ldots,m_{tree-1}\},i\in D_{oob,t}}\left(\hat{\sigma}^2_{oob.iv,i,t}\right)$. Thinking about m_{leaf} again, we can expect $\hat{\sigma}^2_{oob.iv}$ to increase in m_{leaf}, since it is easier to group similar event histories when many groups are available for the task.

$\hat{\sigma}^2_{oob.mean}$ and $\hat{\sigma}^2_{oob.iv}$ can help us understand what happens to the average tree in the ensemble, but it does not help us much in choosing m_{tree} and in understanding more precisely how the global model fit changes in m_{try} and $m_{leaf.}$

An approach to global fit evaluation could be to examine how well the estimated lifetime value $\hat{v}$ ranks observations with respect to observed average revenue per time unit $\hat{a}_i = \left(\sum_{j=0}^{n_i-1} r_{s_{ij}}\right)/n_i$, by some measure. However, $\hat{\alpha}_i$ is usually a weak estimator for the true average revenue, since the event history is likely to be short. Other alternative measures could be developed, but many obvious candidates suffer from the need of an individual level estimation of the markov transition matrix. This could clearly be done in a tree ensemble, but the convergence properties of even more complicated objects would then be an issue.

3.1.2 Variable Importance

Several measures to judge variable importance are used in the tree ensemble literature and code base. The term importance, and not significance, is used to emphasize that the primary objective is to know if an attribute matters when explaining the variational patterns in an outcome variable, not how it matters in the sense of directional influence. The importance rankings should preferably be significant, but some measures with unknown statistical properties are used as a guiding tool for exploratory analysis anyway.

A measure used primarily as a guiding tool is easily calculated as a by-product of tree generation: The average change in $w(\cdot)$, in eq. 7, from splitting on a variable, averaged over all trees.

A second measure with a degree of statistical validity is based on a permutation approach. Given a local measure of out-of-bag fit ρ, for each attribute, the measure is calculated by doing one or more random permutations of the attribute in every tree and record the resulting change in ρ. The t-score for the mean change can then be calculated by observing the mean, standard deviation and number of trees. Note that ρ is a measure of fit with respect to a single tree, not a measure of how the trees fit collectively. In this paper ρ is set to a pseudo chi-square test, a form of average chi-square taken over all observations in a leaf, using the out-of-bag means instead of some observed quantity as in the actual chi-square.

$$\rho = \hat{E}\frac{\left(E\left(v_{\hat{M}_{D_{in}},s_{i\theta}}\right) - E\left(v_{\hat{M}_{D_{oob}},s_{i\theta}}\right)\right)^2}{\frac{1}{|D_{in}|}\hat{\sigma}^2_{in.iv,i,t} + \frac{1}{|D_{oob}|}\hat{\sigma}^2_{oob.iv,i,t}} \quad (11)$$

The averaging is over all out-of-bag observations in a tree, and D_{in} and D_{oob} is implicitly understood to vary with each leaf as described in the algorithm-section.

3.1.3 Segmentation

Breiman (2001) suggests a novel and interesting way of evaluating observation proximities in tree ensembles. The idea is to count how often observations end up in the same leaf, then transform this information into a matrix of dissimilarity measures. Finally, multidimensional scaling, as described in Cox & Cox (1994), can be applied, to visualize proximities.

Initialize the $n_D \times n_D$ matrix F to zeroes and let $F_{i,j}$ denote element (i,j) in F. In function tree_leaf$(\cdot)$ we extend the leaf updating mechanism as stated below.

for each $i \in D_{oob}$:
  $\hat{v}_{it} = E\left(v_{\hat{M}_{D_{in}}}\right)$
  for each $j \in D_{oob}$:
    $F_{i,j} = F_{i,j} + 1$

Upon termination of the algorithm, we can now calculate a dissimilarity matrix F^{dis} as in eq. 12.

$$F^{dis} = I_{n_D \times n_D} - diag(F)^{-1} F \quad (12)$$

Each entry in F^{dis} is guaranteed to reside in the unit interval. A low value of $F_{i,j}$ can be interpreted as if observation i and j are similar with respect to attributes that carry importance in determining outcome.

For larger values of n_D some special precautions are necessary when performing these computations. Firstly, some multidimensional scaling algorithms are unable to handle n_D's ranging in the thousands. Secondly, a $n_D \times n_D$ matrix of say, 4 byte integer values or 8 byte real values, eats up memory. F^{dis} for $n_D = 10000$ would consume $(10000^2 \times 8)/1024^2 \approx 763$ megabytes of memory. The solution used by this author is to use a sparse matrix memory layout and then sample a low number of rows and columns to be used by the multidimensional scaling package. Another solution would be to mark a small random subsample of D for proximity calculation and then count leaf memberships for this sample alone.

3.1.4 Speed of Algorithm and Search Resolution

If D is large and $\mathcal{S}$ contains many elements, then it can be very time-consuming to complete a search sweep for the best split. Just one sweep along a single dimension would involve inverting a $|\mathcal{S}| \times |\mathcal{S}|$ matrix up to n_D times. A way to mitigate this is to coarsen the search resolution. In the tree_branch($\cdot$) function, instead of varying y such that $y \in \{y_{ik} | i \in D_{in}\}$, we can vary y over every $m_{resolution}$'th element in the range, thus bringing the necessary number of evaluations down from $|\{y_{ik} | i \in D_{in}\}|$ to the nearest $\frac{|\{y_{ik}|i \in D_{in}\}|}{m_{resolution}}$. The gain in speed comes at the cost of decreased precision, but different values can be tried out on small tree ensembles to judge the effects in each case.

A second solution is to increase the available processing power. Tree ensembles are very easy to run in parallel on more processors, since each tree does not need information from any other tree during generation.

3.1.5 Empty Cells - A Bayesian Solution

When partitioning D into smaller groups, there is a risk that eq. 5 is undefined since

$$\sum_{i \in \hat{D}} \sum_{j=1}^{n_i - 1} 1\left(s_{i,j-1} = k\right) = 0.$$

This happens when agents in a partition never occupy state k. In some scenarios this could have perfectly reasonable explanations due to some inherent properties of the studied phenomena. In such cases it would be preferable to explicitly build such knowledge into the model, but that is outside the scope of this paper.

In other scenarios the empty cells could arise from a low probability of entering the state, either from other states or as an initial state. The approach taken in this paper is to use a Bayesian weighing scheme, combining aggregate information about behavior from a transition probability matrix M^{prior} with local information in each leaf. A new step is entered in function tree_root($\cdot$), letting $M^{prior} = \hat{M}_{D_{in}}$. Given a prior weight $\alpha > 0$, the procedure used in calculating the group-level transition probability matrix is changed from eq. 5 to eq. 13. Note that for $\alpha = 0$ the latter equation collapses to the former.

$$\hat{M}_{\hat{D}} = \left\{ \frac{\left(\sum_{i \in \hat{D}} \sum_{j=1}^{n_i-1} 1\left(s_{i,j-1} = k \text{ and } s_{i,j} = l\right)\right) + \alpha \left|\hat{D}\right| M_{k,l}^{prior}}{\left(\sum_{i \in \hat{D}} \sum_{j=1}^{n_i-1} 1\left(s_{i,j-1} = k\right)\right) + \alpha \left|\hat{D}\right|} \right\}_{(k,l) \in \mathcal{S} \times \mathcal{S}}, \qquad (13)$$

$M_{k,l}^{prior}$ denoting element (k, l) from M^{prior}. The entries of M^{prior} is weighted with the number of observations in the group, to distribute the mass of the prior over all leaves in the tree. Failing to weight by $\left|\hat{D}\right|$ would result in a skewing of the search in the sense that each new leaf entering would introduce more probability mass in the model. Typically α would be set in proportion to $\frac{1}{|\hat{D}|}$.

3.1.6 Censoring

The Markov property of the model helps in dealing with random censoring of observations. For an event history $h_i = \{s_{i,0}, s_{i,1}, .., s_{i,n_i-1}\}$, $s_{ij} = -1$ if it is censored, we form a sequence of observed transitions $h_i' = \{(s_{i,j-1}, s_{i,j}) \,|s_{i,j-1} \neq -1 \text{ and } s_{i,j} \neq -1\}$ and modify eq. 5 to

$$\hat{M}_{\hat{D}} = \left\{ \frac{\sum_{i\in\hat{D}} \sum_{j\in h_i'} 1\,(j_0 = k \text{ and } j_1 = l)}{\sum_{i\in\hat{D}} \sum_{j\in h_i'} 1\,(j_0 = k)} \right\}_{(k,l)\in\mathcal{S}\times\mathcal{S}}. \tag{14}$$

Here j_i is element i in the tuple. If h_i' is empty we are forced to ignore the observation, since it carries no information with respect to transitions.

If $s_{i,0}$ is censored we cannot calculate eq. 6 correctly, but can instead ignore the censored observations with respect to their initial data. This introduces the possibility of having a group of observations all having censored initial states. In that case, as in the section above, a Bayesian approach is utilized based on the aggregate D_{in} in the function tree_root($\cdot$).

3.1.7 State-specific Discount Rates

In the application section we will study a scenario with states of varying duration. This has implications for how a discount rate is incorporated, since revenue following a state of a long duration must be discounted more than the opposite situation. Let g_i denote the time unit duration of state i. For a discount factor β expressed in the same time units, let the $|\mathcal{S}| \times |\mathcal{S}|$ diagonal matrix $G = diag\left(\{\beta^{g_i}\}\right)$. It can be shown that the former results regarding internal variance and expected lifetime value holds, if we exchange in eq. 3 β for G and modify eq. 4 as in eq. 15.

$$\begin{aligned} var\,(v) &= (I - GMG')^{-1}\,\theta, \\ \theta &= M\,(r + Gv) \circ (r + Gv) - v \circ v \end{aligned} \tag{15}$$

3.2 Simulation

Let us now return to our imaginary newspaper scenario. Suppose instead of a fixed markov transition matrix, as on p. 82, that probabilities vary with some observed attributes while the revenue and discount factor are unchanged.

3.2.1 Linear Scenario

For agent i, suppose that attribute j, x_{ij} lies on a unit interval. Suppose also that a markov transition probability matrix M_i varies with the first two of k attributes as in eq. 16.

$$M_i = \begin{bmatrix} & 0 & \mathsf{R} & \mathsf{F} \\ 0 & 0.8 & 0.2 & 0 \\ \mathsf{R} & 1 - x_0 & 0 & x_0 \\ \mathsf{F} & 1 - x_1 & 0 & x_1 \end{bmatrix} \tag{16}$$

x_0 could be interpreted as the propensity to enter into a full-price subscription, while x_1 is a loyalty dimension.

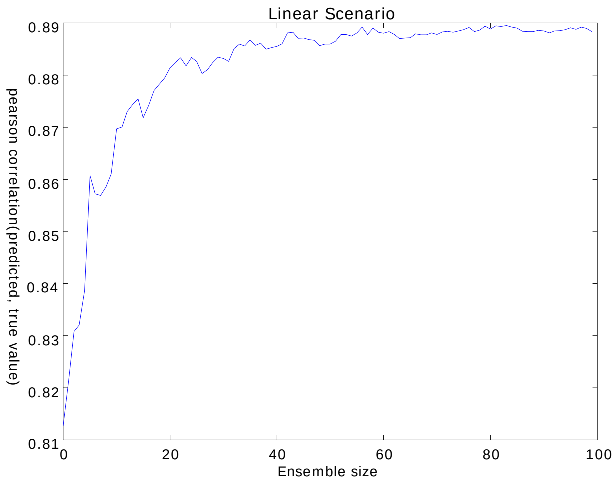

Figure 1: Linear scenario. Pearson correlation of estimated ltv and true ltv as a function of ensemble size

A simulation with $n = 1000$ such agents are generated, with $k = 10$ attributes. 20 events are simulated for each agent and the initial distribution is picked at random. The true expected lifetime value is recorded and compared to the estimated.

An ensemble with $m_{try} = 8$ and $m_{leaf} = 70$ is grown. The correlation between estimated and expected value as a function of ensemble size is shown in figure 1 p. 91. Variable importance is showed in figure 2 p. 92. It is clear that x_0 and x_1 are dominant, as expected.

The proximity matrix of 500 agents is sampled and two compontens are extracted via multidimensional scaling. The plot is seen in figure 3 p. 92. Colors represent an index of the estimated lifetime value over the full sample's average lifetime value. Three distinct segments seem to exist, representing sets of (x_0, x_1). The most valuable fifth, the white segment, has a median x_0, x_1 of 0.73 and 0.91. The least valuable segment, the black segment, has a median x_0 and x_1 of 0.19 and 0.37. The middle segment has a median x_0 of 0.83 and $x_1$of 0.73, so they enter into full-time subscription, but only for at short period.

3.2.2 Non-linear Scenario

Let us now introduce a non-linear complication. Define

$$M_i^{opposite} = \begin{bmatrix} & 0 & R & F \\ 0 & 0.8 & 0.2 & 0 \\ R & x_0 & 0 & 1 - x_0 \\ F & x_1 & 0 & 1 - x_1 \end{bmatrix},$$

and let

$$M_i' = \begin{cases} M_i & \text{if } x_{i2} \leq \frac{1}{2} \\ M_i^{opposite} & \text{if } x_{i2} > \frac{1}{2} \end{cases}.$$

M_i' is used as the new transition probability matrix when simulating, but everything else is set up as in the previous section. As seen from figure 4 p. 93, the convergence in pearson correlation is slower and reaches a plateau at a lower level. It is more difficult to predict correctly due to the non-linearity.

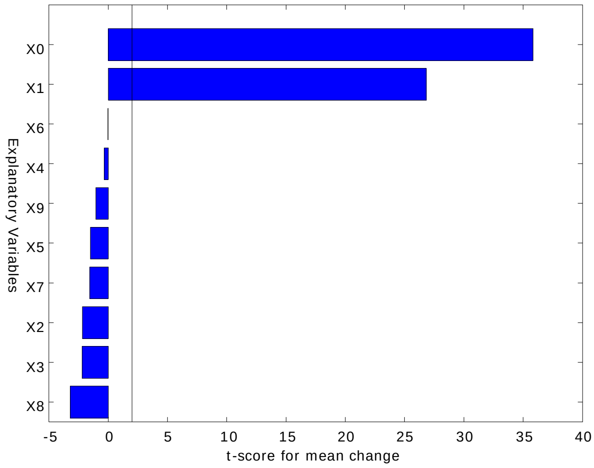

Figure 2: Linear scenario. Variable importance as t-score for mean change in pseudo chi-square measure

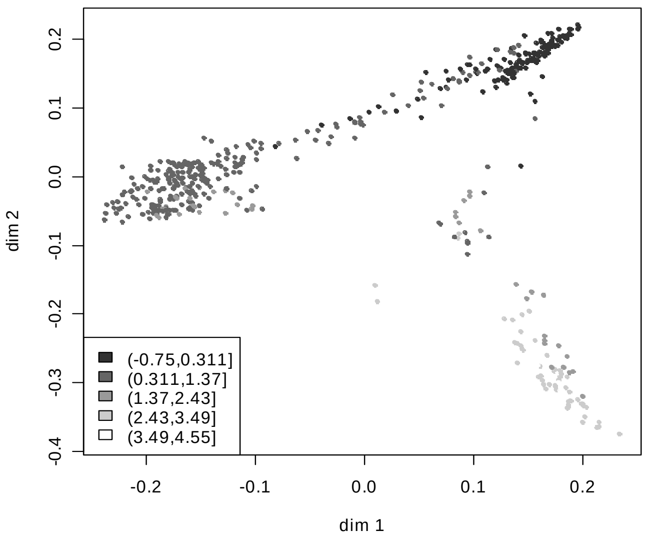

Figure 3: Plot of two dimensions from a multidimensional scaling of the linear scenario proximity matrix. Colors represent est. ltv over average ltv

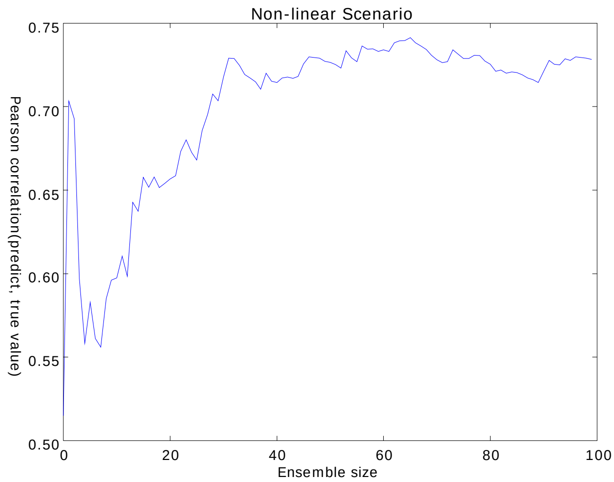

Figure 4: Non-linear scenario. Pearson correlation of estimated ltv and true value as a function of ensemble size

The variable importance plot picks out the correct variables again, as seen in figure 5 p. 5. Many simple linear models would not have found x_0 and x_1 significant in this case.

The segmentation plots, shown in figures 6 and 7, are constructed with three components, since a two-component plot was more difficult to interpret. In figure 7 we see a white cluster of high-value customers again. Those are customers with either very high or very low values of x_0 and x_1, depending on the value of x_2. The segmentation is a bit more difficult to visualize, due to the non-linearity.

All in all, the simulations indicate that the model could be useful in the face of real data.

4 Application

In this section the subscriber transaction history database of a daily newspaper is explored. The goal is to learn how customer lifetime value derived from transitions through a relationship statespace varies across customer attributes and possibly identify distinct customer segments. The relationship statespace consists of various introductory offers that lead to states of full-price subscription. The introductory offers are provided at a cost to the company or yield little revenue, while full price states generate the majority of revenue.

Due to the nature of the data, we do not observe agents before they enter a subscription program. Conclusions derived from this study is thus conditional on the subset of all agents willing to enter into a least one type of subscription at some point in time.

The newspaper is anonymous to avoid handing out sensitive information. It can be characterized as a mainstream media outlet with a circulation that puts it among some of the largest in it's country. Besides a core section of general news being published every day, extra sections about food, cars, fashion and so on is added at various fixed days of the week. The weekend edition is special, in that many extra sections are available to the reader. The paper is sold at newsstands and on a subscriber basis, but this study will concentrate on the subscriber part only.

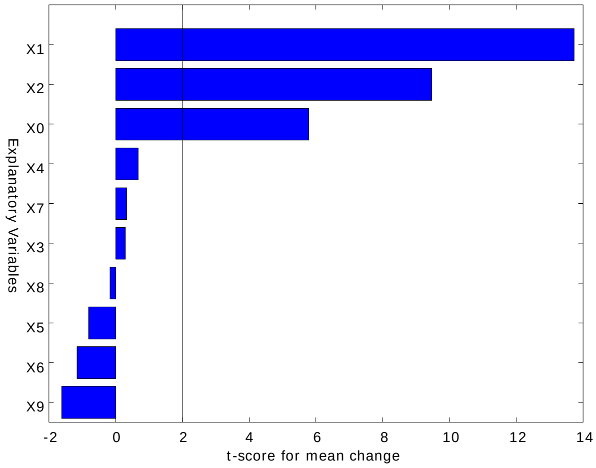

Figure 5: Non-linear scenario. Variable importance measured as t-score for change in pseudo chi-square measure

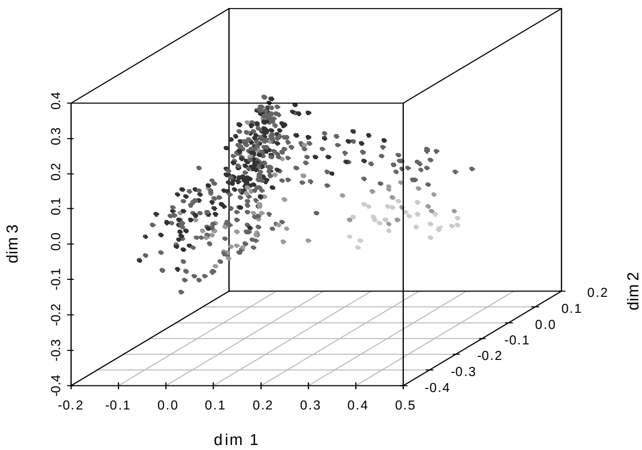

Figure 6: Non-linear scenario. Three components extraced from the proximity matrix with multidimensional scaling

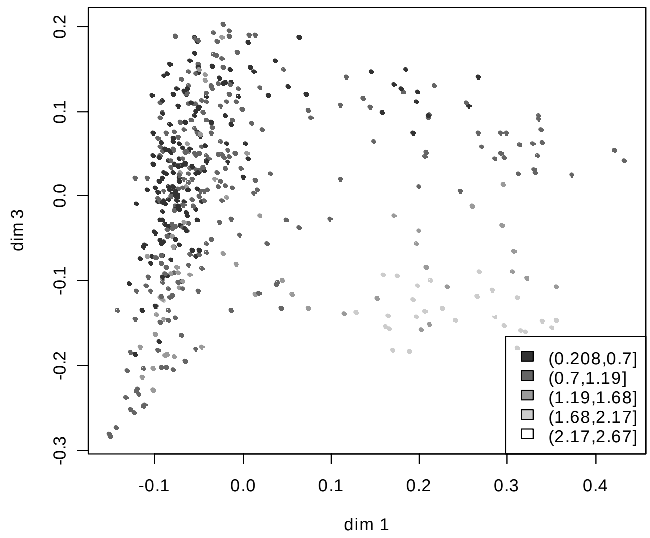

Figure 7: Non-linear scenario. Plot of component 1 and 3 from multidimensional scaling of proximity matrix.

4.1 Data

10,237 current or former customer event histories are sampled from the newspaper's database. 1^{st} of January 1998 is set as the starting date for an observation to be eligible for sample inclusion, while the15^{th} of January 2006 marks the ending date of any observation. The mean duration of an event history is 29.3 months and the median is 27 months, including the non-customer as an event.

A fine-grained commercial database provides address-linked sociodemographic information about customers. The information is provided on varying group levels due to laws, and the practice of laws, that regulate the commercial use of of person data registers. They require varying minimum calculation group sizes as some data elements are perceived as more sensitive than others. In general, very sensisitive statistics require a minimum of 150 households, while less sensitive items require in the range of 50 households. The commercial database links customer addresses to flexible map polygons that meet the required demands. Customers are then assigned the aggregate statistics relating to all households within the polygon.

An overview of all 82 attributes are listed in figure 8. Nearly all variables are in percentages i.e. jt_agri =5 means that 5% of all people in the local polygon are employed in jobs related to agriculture.

To get a feeling for a typical event history 50 agents are sampled at random. Their event histories are plotted in figure 9. Each line represents the event history of a single agent, from the time he first entered into any type of relationship with the newspaper, to the end of the observation window. The varying lengths of event histories are a function of the starting date: Later starting dates result in short event histories. Figure 9 illustrates how most agents start out by receiving an introductory discount on their subscription. The typical discount for an one-month introductory subscription is around 50%, while discounts for longer subscriptions usually range around 35% on the full price.

Map Coordinates	x	Map x-coordinate
	y	Map y-coordinate
	cap_dist	Distance from capital
Age	age_0_11	Percentage of group
	age_12_16	aging from a to b,
	age_17_22	shown as _a_b and
	age_23_29	66p as 66 years and
	age_30_39	above
	age_40_49	
	age_50_59	
	age_60_65	
	age_66p	
Job Status	js_selfemp	Self-employed
	js_basic	Basic, medium,
	js_mid	high and top-leader
	js_hi	jobs according to
	js_topleader	responsibility.
	js_cashbenefit	Public benefit
	js_erl_ret	Early Retired
	js_j_rel_pen	Job-release Pension
	js_student	Student
	js_pension	Pensioner
	js_na	No information
	js_child	Children, not working
Building Types	bt_agri	Farm buildings
	bt_house	Houses
	bt_mult	Multi-houses
	bt_terrace	Terraced houses
	bt_apts	Apartments
	bt_na	No information
Family Type	f_sic	Single income w/child
	f_sinc	Single income no child
	f_dic	Double income w/child
	f_dinc	Double income no child
	f_na	No information
Household Wealth	w_0_1	Percentage households
	w_1_5	with wealth in national
	w_5_25	[a,b] percentile
	w_25_50	
	w_50_75	
	w_75_90	
	w_90_95	
	w_95_99	
	w_99_100	

Education	edu_primary	Primary School
	edu_gym	Gymnasium
	edu_egym	Business Gymnasium
	edu_crft	Craft
	edu_short	Short education
	edu_medium	Medium education
	edu_bach	Bachelor degree
	edu_grad	Graduate degree
	edu_na	No information
Job Type	jt_agri	Agriculture
	jt_manufact	Manufacturing
	jt_supply	Utilities
	jt_construct	Construction
	jt_trade	Trade, hospitality
	jt_transport	Transport
	jt_finance	Finance
	jt_public	Public and personal services
	jt_na	No information
	jt_child	Children
	jt_unempl	Unemployed
	jt_pension	Pension
Ownership Types	ot_owner	Owned
(house ownership)	ot_privaterent	Rented, private
	ot_publicrent	Rented, publiv
	ot_coop	Cooperative
Building Sizes	bs_40b	Below 40m2
	bs_40_70	
	bs_71_90	
	bs_91_130	
	bs_130p	130m2 or more
Household Income	inc_b33	Below $33k
	inc_33_75	
	inc_75p	75k$ or more
Personal Income of Highest Household Earner	pinc_b20	Income in national
	pinc_20_40	[a,b] percentile
	pinc_40_60	
	pinc_60_80	
	pinc_80_100	

Figure 8: Variable overview and explanation.

Figure 9: A random sample of event histories. Yellow: Discount subscription, green: Full-price subscription, red: no subscription, grey: unobserved.

4.2 Relationship Statespace

Let us now construct a relationship statespace that models key customer events. States are constructed to take into account if a customer is participating in some kind of discount program or if he is a regular customer paying the full price. Other properties of a customer relationship state include the length of the subscription term, a measure of commitment, and how often the newspaper is delivered.

The statespace is a simplified representation of the true possibilities facing a customer. For instance, one can choose to get nothing but the Sunday edition of the newspaper, but in the model it is only possible to choose between a daily and a weekend edition. The subscription term may also vary beyond the given option of three months and twelve months, but most customers happen to fall in the mentioned categories. Event histories that don't fit the statespace are modified to fit in. As an example, a customer with a six month full-price subscription is simply chopped into two three-month subscriptions. The statespace is simplified to minimize the computational and data degrees of freedom cost, while also easing interpretation of later model output.

The names of relation states are coded to minimize the size of each state symbol in graphs and diagrams.

4.2.1 State Codes

Pricing Scheme. A customer is either participating in an introductory pricing program, is paying the full price or receives some other type of rebate, such as a loyalty rebate.

- I Introductory Rebate
- F Full Price
- R Non-introductory Rebate

Product type. A customer is either subscribing for the daily newspaper or the weekend edition.

- D Daily
- W Weekend Edition

Duration of commitment. An introductory rebate might run for around a month, a quarter or a full year. Full price subscription states are simplified to run over a quarter or a year only.

- M Comitted for a month
- Q Quarter
- B Biannually
- Y Comitted for a full year

Duration of stay in full-price states. The number of consecutive times the customer pays the full price is counted up to a limit. Example: State "FDY.2" means that a customer is subscribing for a daily, with a commitment running over a year. He has been paying the full price for 2 years in a row. State "FWQ.3" is for a customer subscribing

State	Revenue	State	Revenue
FDQ.1	100.00	FWQ.1	46.00
FDQ.2	100.00	FWQ.2	46.00
FDQ.3	100.00	FWQ.3	46.00
FDQ.4	100.00	FWQ.4	46.00
FDY.1	373.00	FWY.1	169.00
FDY.2	373.00	FWY.2	169.00
FDY.3	373.00	FWY.3	169.00

State	Revenue	State	Revenue
IDB	22.00	IWB	-20.00
IDM	-6.00	IWM	-3.00
IDQ	13.00	IWQ	-1.00
IDY	34.00	IWY	-11.00
RDQ	45.00	RWQ	21.00
RDY	181.00	RWY	84.00
		0	0.00

Figure 10: Revenue index earned in a given state. Index set to revenue from a full-price daily subscription in a quarter.

for the weekend edition with the commitment running over a quarter. A customer in this state has been paying the full price for 3 quarters in a row. States with yearly commitments count no more than three years, while quarterly commitment are counted up to four quarters.

Churn state. Former customers enter state "0". State 0 is assumed to last for a quarter. Then it might be re-entered, which is the most common pattern.

4.2.2 Revenue

Figure 10 p. 99 shows the revenue earned when a customer occupy a given state. The numbers are not exact in order to protect the newspaper, but the relative magnitude is true to the actual values. Since all states represent a simplification of reality, the revenue numbers are modified to mirror this. As an example, quarterly revenue on weekend edition subscriptions are taken as a weighted average of Fri-Sat-Sun subscribers and Sunday subscribers alone. The weights set to reflect the relative frequency of the various subscription forms.

We will think of the 100 units earned from a full-price daily in a quarter, FDQ, as $100. It is clear that the company earns more from daily subscribers, than from weekend edition subscribers. Introductory rebates are higher than non-introductory rebates, so less money is earned from the former. For introductory rebates, we see that weekend edition discounts are handed out at a loss, while introductory daily discounts still earns the company a bit of revenue.

State	n	p
FDQ.1	217	0.026
FDY.1	53	0.006
FWQ.1	109	0.013
IDB	953	0.112
IDM	216	0.025
IDQ	3712	0.438
IDY	1157	0.136
IWB	459	0.054
IWM	452	0.053
IWQ	977	0.115
IWY	175	0.021
Sum	8480	1

Figure 11: Initial state absolute frequency n and relative frequency p

4.3 Aggregate Statistics

The first state an agent is seen in, in the customer database, is interpreted as his initial state. The initial state is used to weight the internal variance of lifetime value in eq. 7, but is also interesting in it's own right. The lifetime value is governed by the transition probability matrix and the initial distribution, so if the initial distribution varies with agent attributes it could be an important source of variation of the lifetime value between segments.

The initial distribution is shown in figure 11. Relative to the total number of observations, about 17% are unobserved at the initial state.

From the table it is clear that the notion about most agents starting out in a relationship state of some introductory offer is correct. The most frequent initial state is that of a three month introductory rebate on a daily newspaper, while the second most and much less frequent state is an introductory rebate on a daily for a full year.

The aggregate transition probability matrix is shown in figure 12 p. 101. It is unwieldy and many entries are equal to zero, so an alternative representation is shown in figure 13.

A second visual representation of transition probabilities in the form of a graph is shown in figure 14 p. 103. Transition probabilities below .05 are not shown in the figure, but their weights are still counted, therefore some outgoing weights will not sum to one. State 0 is not represented directly to avoid cluttering the figure, but the probability of a transition to state 0 is shown as a number below each state label.

Studying the graph closer reveals some interesting paths followed by customers. Take as a point of departure state IDQ, the most frequent state. From here around 0.4 churns, while 0.54 move on to a full-price daily subscription (FDQ.1) for a quarter. From here, the majority 0.57 churns, while 0.39 enter into a second quarter of full-price subscription. From FDQ.2, again, the majority churns, but 0.31 continue to a third quarter. In FDQ.3 the churn rate declines considerably and 0.83 move on to a fourth quarter. In FDQ.4 agents have even lower rates of churn, while 0.91 continues from quarter to quarter. This is a key path with a close link to the revenue-dynamics of the newspaper.

A similar typical path for the second most frequent initial state,is that, in short form, of (IDY, FDY.1, RDY, FDY.1, FDY.2, FDY.3). After a year of full-price subscription, many customers get a loyalty-discount and return to a full-pricing scheme and in time, ending up in a relatively stable state of repeat-subscription from year 3 and onwards.

	0	FDQ.1	FDQ.2	FDQ.3	FDQ.4	FDY.1	FDY.2	FDY.3	FWQ.1	FWQ.2	FWQ.3	FWQ.4	FWY.1	FWY.2
0	0.968	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
FDQ.1	0.570	0.000	0.389	0.000	0.000	0.000	0.002	0.000	0.000	0.008	0.000	0.000	0.000	0.000
FDQ.2	0.565	0.000	0.000	0.310	0.000	0.000	0.008	0.000	0.000	0.000	0.010	0.000	0.000	0.000
FDQ.3	0.140	0.000	0.000	0.000	0.809	0.000	0.008	0.000	0.000	0.000	0.000	0.003	0.000	0.000
FDQ.4	0.068	0.000	0.000	0.000	0.902	0.000	0.013	0.000	0.000	0.000	0.000	0.003	0.000	0.000
FDY.1	0.377	0.000	0.000	0.000	0.010	0.000	0.240	0.000	0.000	0.000	0.000	0.000	0.000	0.002
FDY.2	0.142	0.000	0.000	0.000	0.015	0.000	0.000	0.722	0.000	0.000	0.000	0.000	0.000	0.000
FDY.3	0.080	0.000	0.000	0.000	0.007	0.000	0.000	0.868	0.000	0.000	0.000	0.000	0.000	0.000
FWQ.1	0.432	0.000	0.002	0.000	0.000	0.000	0.000	0.000	0.000	0.542	0.000	0.000	0.000	0.001
FWQ.2	0.513	0.000	0.000	0.003	0.000	0.000	0.000	0.000	0.000	0.000	0.402	0.000	0.000	0.004
FWQ.3	0.145	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.814	0.000	0.003
FWQ.4	0.109	0.000	0.000	0.000	0.001	0.000	0.000	0.000	0.000	0.000	0.000	0.858	0.000	0.005
FWY.1	0.408	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.281
FWY.2	0.239	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
FWY.3	0.077	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.019	0.000	0.000
IDB	0.235	0.679	0.000	0.000	0.000	0.005	0.000	0.000	0.002	0.000	0.000	0.000	0.002	0.000
IDM	0.811	0.068	0.000	0.000	0.000	0.008	0.000	0.000	0.004	0.000	0.000	0.000	0.000	0.000
IDQ	0.395	0.541	0.000	0.000	0.000	0.001	0.000	0.000	0.004	0.000	0.000	0.000	0.000	0.000
IDY	0.177	0.061	0.000	0.000	0.000	0.639	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IWB	0.269	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.648	0.000	0.000	0.000	0.002	0.000
IWM	0.827	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.051	0.000	0.000	0.000	0.000	0.000
IWQ	0.314	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.593	0.000	0.000	0.000	0.000	0.000
IWY	0.227	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.033	0.000	0.000	0.000	0.603	0.000
RDQ	0.158	0.471	0.000	0.000	0.000	0.032	0.000	0.000	0.049	0.000	0.000	0.000	0.016	0.000
RDY	0.018	0.101	0.000	0.000	0.000	0.784	0.000	0.000	0.005	0.000	0.000	0.000	0.000	0.000
RWQ	0.137	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.584	0.000	0.000	0.000	0.000	0.000
RWY	0.031	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.124	0.000	0.000	0.000	0.465	0.000

	FWY.3	IDB	IDM	IDQ	IDY	IWB	IWM	IWQ	IWY	RDQ	RDY	RWQ	RWY
0	0.000	0.006	0.001	0.006	0.008	0.003	0.001	0.003	0.001	0.000	0.000	0.000	0.000
FDQ.1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.020	0.007	0.001	0.000
FDQ.2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.061	0.031	0.003	0.003
FDQ.3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.010	0.005	0.000	0.000
FDQ.4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.004	0.005	0.000	0.000
FDY.1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.024	0.327	0.001	0.007
FDY.2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.012	0.065	0.004	0.000
FDY.3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.021	0.000	0.000
FWQ.1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.002	0.001	0.011	0.002
FWQ.2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.003	0.055	0.006
FWQ.3	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.006	0.000
FWQ.4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.001	0.001	0.013	0.004
FWY.1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.007	0.028	0.204
FWY.2	0.498	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.060
FWY.3	0.689	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.019
IDB	0.000	0.057	0.003	0.004	0.002	0.003	0.000	0.000	0.000	0.000	0.000	0.000	0.000
IDM	0.000	0.004	0.004	0.053	0.000	0.008	0.004	0.000	0.000	0.000	0.000	0.000	0.000
IDQ	0.000	0.000	0.001	0.051	0.001	0.000	0.000	0.002	0.000	0.000	0.000	0.000	0.000
IDY	0.000	0.002	0.002	0.001	0.109	0.000	0.001	0.000	0.001	0.000	0.000	0.000	0.000
IWB	0.000	0.000	0.000	0.002	0.000	0.053	0.000	0.005	0.002	0.000	0.000	0.000	0.000
IWM	0.000	0.000	0.000	0.000	0.002	0.002	0.006	0.090	0.004	0.000	0.000	0.000	0.000
IWQ	0.000	0.000	0.000	0.000	0.001	0.002	0.027	0.054	0.001	0.000	0.000	0.000	0.000
IWY	0.000	0.000	0.004	0.000	0.012	0.004	0.000	0.000	0.074	0.000	0.000	0.000	0.000
RDQ	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.134	0.089	0.008	0.000
RDY	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.005	0.041	0.000	0.000
RWQ	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.103	0.000
RWY	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.062

Figure 12: Aggregate transition probability matrix

Source state(n): (top_i_destination_state: probability)
0 (48078): (0 : 0.968) (IDY : 0.008) (IDB : 0.006) (IDQ : 0.006) (IWB : 0.003)
FDQ.1 (3371): (0 : 0.571) (FDQ.2 : 0.390) (RDQ : 0.020) (FWQ.2 : 0.008) (RDY : 0.007)
FDQ.2 (1168): (0 : 0.570) (FDQ.3 : 0.313) (RDQ : 0.062) (RDY : 0.032) (FWQ.3 : 0.010)
FDQ.3 (383): (FDQ.4 : 0.830) (0 : 0.144) (RDQ : 0.010) (FDY.2 : 0.008) (RDY : 0.005)
FDQ.4 (3204): (FDQ.4 : 0.905) (0 : 0.068) (FDY.2 : 0.013) (RDY : 0.005) (RDQ : 0.004)
FDY.1 (970): (0 : 0.381) (RDY : 0.331) (FDY.2 : 0.242) (RDQ : 0.025) (FDQ.4 : 0.010)
FDY.2 (250): (FDY.3 : 0.752) (0 : 0.148) (RDY : 0.068) (FDQ.4 : 0.016) (RDQ : 0.012)
FDY.3 (415): (FDY.3 : 0.889) (0 : 0.082) (RDY : 0.022) (FDQ.4 : 0.007)
FWQ.1 (1295): (FWQ.2 : 0.546) (0 : 0.436) (RWQ : 0.012) (RDQ : 0.002) (FDQ.2 : 0.002)
FWQ.2 (687): (0 : 0.521) (FWQ.3 : 0.408) (RWQ : 0.055) (RWY : 0.006) (FWY.2 : 0.004)
FWQ.3 (308): (FWQ.4 : 0.841) (0 : 0.149) (RWQ : 0.006) (FWY.2 : 0.003)
FWQ.4 (1362): (FWQ.4 : 0.865) (0 : 0.109) (RWQ : 0.013) (FWY.2 : 0.005) (RWY : 0.004)
FWY.1 (132): (0 : 0.439) (FWY.2 : 0.303) (RWY : 0.220) (RWQ : 0.030) (RDY : 0.008)
FWY.2 (40): (FWY.3 : 0.625) (0 : 0.300) (RWY : 0.075)
FWY.3 (42): (FWY.3 : 0.857) (0 : 0.095) (FWQ.4 : 0.024) (RWY : 0.024)
IDB (1120): (FDQ.1 : 0.685) (0 : 0.237) (IDB : 0.057) (FDY.1 : 0.005) (IDQ : 0.004)
IDM (255): (0 : 0.843) (FDQ.1 : 0.071) (IDQ : 0.055) (IWB : 0.008) (FDY.1 : 0.008)
IDQ (4128): (FDQ.1 : 0.542) (0 : 0.396) (IDQ : 0.052) (FWQ.1 : 0.004) (IWQ : 0.002)
IDY (1466): (FDY.1 : 0.643) (0 : 0.178) (IDY : 0.110) (FDQ.1 : 0.061) (IDM : 0.002)
IWB (539): (FWQ.1 : 0.660) (0 : 0.275) (IWB : 0.054) (IWQ : 0.006) (IDQ : 0.002)
IWM (524): (0 : 0.844) (IWQ : 0.092) (FWQ.1 : 0.052) (IWM : 0.006) (IWY : 0.004)
IWQ (1183): (FWQ.1 : 0.598) (0 : 0.317) (IWQ : 0.055) (IWM : 0.027) (IWB : 0.002)
IWY (232): (FWY.1 : 0.629) (0 : 0.237) (IWY : 0.078) (FWQ.1 : 0.034) (IDY : 0.013)
RDQ (236): (FDQ.1 : 0.492) (0 : 0.165) (RDQ : 0.140) (RDY : 0.093) (FWQ.1 : 0.051)
RDY (208): (FDY.1 : 0.822) (FDQ.1 : 0.106) (RDY : 0.043) (0 : 0.019) (RDQ : 0.005)
RWQ (48): (FWQ.1 : 0.708) (0 : 0.167) (RWQ : 0.125)
RWY (22): (FWY.1 : 0.682) (FWQ.1 : 0.182) (RWY : 0.091) (0 : 0.045)

Figure 13: Aggregate group. Top 5 transition probabilities from source state (first entry in row) to destination state (entry two and onwards).

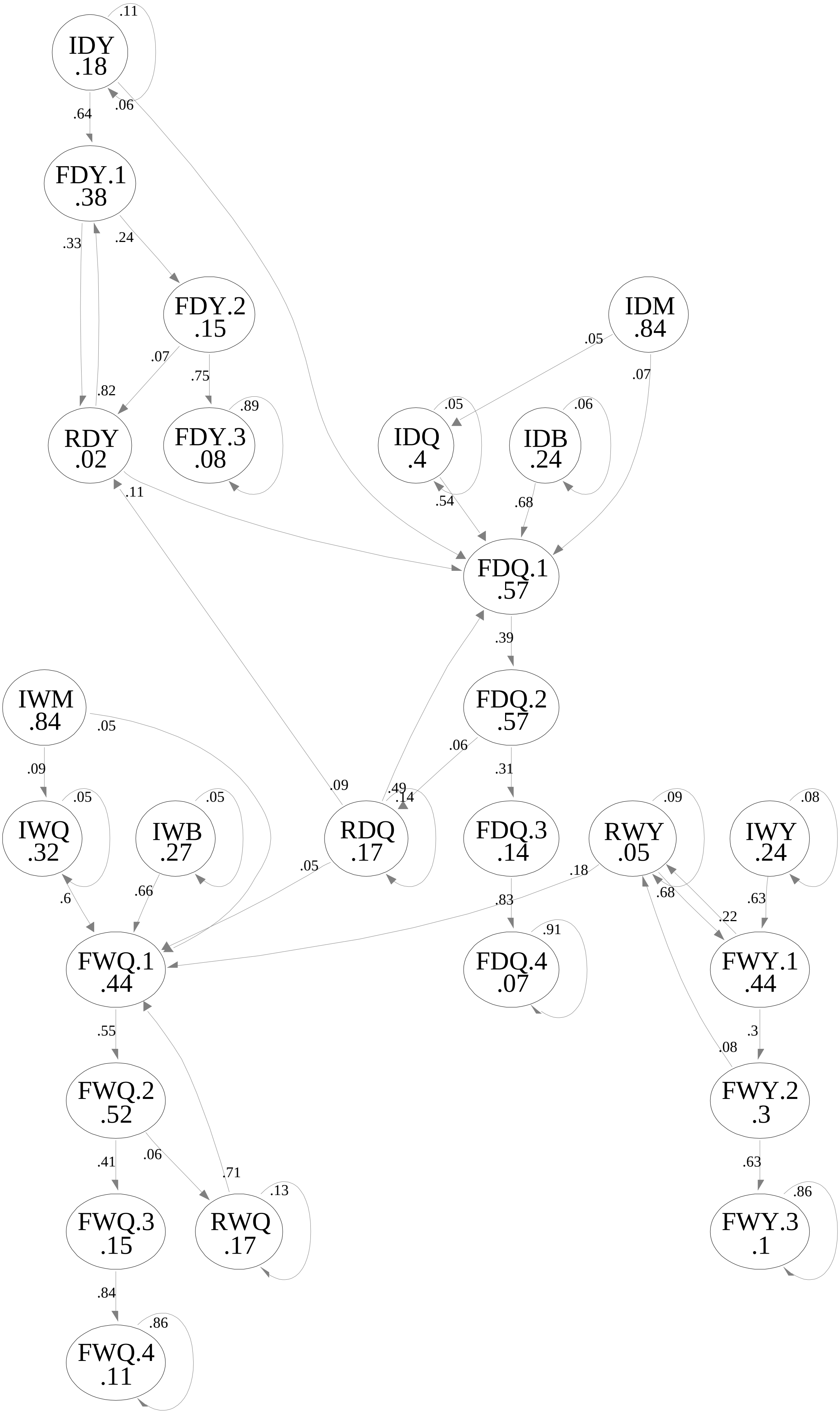

Figure 14: Graph of agent behavior

State	E(v)	E(v)-E(v_0)	State	E(v)	E(v)-E(v_0)
FDQ.1	492	168	FWQ.1	384	60
FDQ.2	644	320	FWQ.2	406	82
FDQ.3	1113	789	FWQ.3	501	177
FDQ.4	1218	894	FWQ.4	523	199
FDY.1	934	610	FWY.1	347	23
FDY.2	1538	1214	FWY.2	358	35
FDY.3	1733	1409	FWY.3	410	86
IDB	498	175	IWB	370	47
IDM	344	20	IWM	327	3
IDQ	468	144	IWQ	378	54
IDY	960	637	IWY	409	85
RDQ	587	263	RWQ	323	-1
RDY	1062	738	RWY	304	-20
			0	324	0

Figure 15: Expected lifetime value by state for the full sample

4.3.1 Aggregate Lifetime Value

The discount rate is set to 10% per year, so $\beta = 0.9$. The expected lifetime value for the aggregate group is displayed in figure 15 p. 104. Column one shows the expected lifetime value, while column two shows the difference down to the expected value of former customers. It could seem counter-intuitive that state 0 customers should have a high expected lifetime value, but here it is important to remember that the analysis is conditional on current and former customers. Former customers might have an expected value of $324, but it would be outside the bounds of the model to conclude that non-customers, the rest of the world, would earn a similar amount.

It is interesting to note that the expected value increases with loyalty. For instance, customers in FDY.2 can be expected to earn $1538 - 934 = 604$ more in present value than FDY.1 customers.

RWQ and RWY, when compared to state 0 customers, carry a negative present value.

4.3.2 Growing a Single Tree

To get a feeling of what the groups might look like, a single tree is grown with $m_{leaf} = 1000$ i.e. no leaf is allowed to contain more than a thousand observations. All variables are permitted to be a candidate at each split, so $m_{try} = 82$. The tree is grown on a random subsample, to permit out-of-bag error-estimates. The result is shown in figure 16 p. 105.

The figure shows the number of observations n in each branch and leaf. For each leaf, the weighted expected lifetime value is reported, the weights governed by the empirical initial state in the leaf-group. An out-of-bag root of the mean squared error is also reported, oob_sd.

The least valuable segment has a wltv of 241, while the most valuable segment is at wltv=951.0, a considerable difference. The least valuable segment is characterized by below average wealth (w_75_90 $<$ 16.34), few working in high job positions and few

```
w_75_90 < 16.34 n=2558
   js_hi < 9.93 n=1789
      bt_house < 49.09 n=1610
         js_job_rel_pen < 1.19 n=315 wltv=241.0 oob_sd=21.86
         js_job_rel_pen >= 1.19 n=1295
            capital_dist < 3761.92 n=259 wltv=251.0 oob_sd=102.59
            capital_dist >= 3761.92 n=1036
               bs_40b < 5.59 n=829 wltv=375.0 oob_sd=32.7
               bs_40b >= 5.59 n=207 wltv=347.0 oob_sd=182.69
      bt_house >= 49.09 n=179 wltv=584.0 oob_sd=182.05
   js_hi >= 9.93 n=769 wltv=383.0 oob_sd=79.9
w_75_90 >= 16.34 n=2561
   w_75_90 < 28.54 n=1535
      pinc_60_80 < 16.60 n=306 wltv=951.0 oob_sd=93.23
      pinc_60_80 >= 16.60 n=1229
         age_23_29 < 2.10 n=118 wltv=640.0 oob_sd=513.44
         age_23_29 >= 2.10 n=1111
            edu_grad < 3.11 n=107 wltv=547.0 oob_sd=159.24
            edu_grad >= 3.11 n=1004
               age_40_49 < 17.99 n=801 wltv=694.0 oob_sd=51.13
               age_40_49 >= 17.99 n=203 wltv=427.0 oob_sd=61.98
   w_75_90 >= 28.54 n=1026
      y < 6128535.00 n=103 wltv=793.0 oob_sd=384.8
      y >= 6128535.00 n=923 wltv=747.0 oob_sd=170.2
```

Figure 16: A partition by a single tree

people on job release pension, a special public pension offered to people in the last phase of their job market participation.

The most valuable segment is characterized by above average wealth (w_75_90 > 16.34), but not way above average (w_75_90 < 28.54). Personal income of highest household earner in the 60-80 percentile is also below average.

In general, the tree tells us to expect wealth and income to be key predictors for customer value, while job position, building types in an area and age plays a significant but smaller role.

4.4 Parameter Selection

The model is run at several parameter values to get an impression of what to expect in terms of out-of-bag mean error and internal variance. m_{try} is allowed to vary over $m_{try} \in \{2, 4, 8, 16, 32, 48, 64, 82\}$ and $m_{leaf} \in \{250,\ 500,\ 1000\}$. The in-bag/out-of-bag sampling rate is set to 0.5. Each pair of parameters are taken through an ensemble of size 25.

Results are shown in figure 17 p. 106. It is seen that small values of m_{try} lead to high internal variance together with relatively high oob mean error. For $m_{try} \geq 48$ there seems to be a kind of efficiency barrier across ensembles of varying m_{leaf} combinations. This barrier probably shows us the optimal trade-off we must do, when choosing between a small internal variance of groups, versus levels of oob mean error.

Without a fixed global fit measure, the authors choice of oob mean error and internal variance is set rather arbitrarily by picking m_{try} = 48 and m_{leaf} = 700. From the figure, one should expect average internal variance near 520 and oob.mean error near 150.

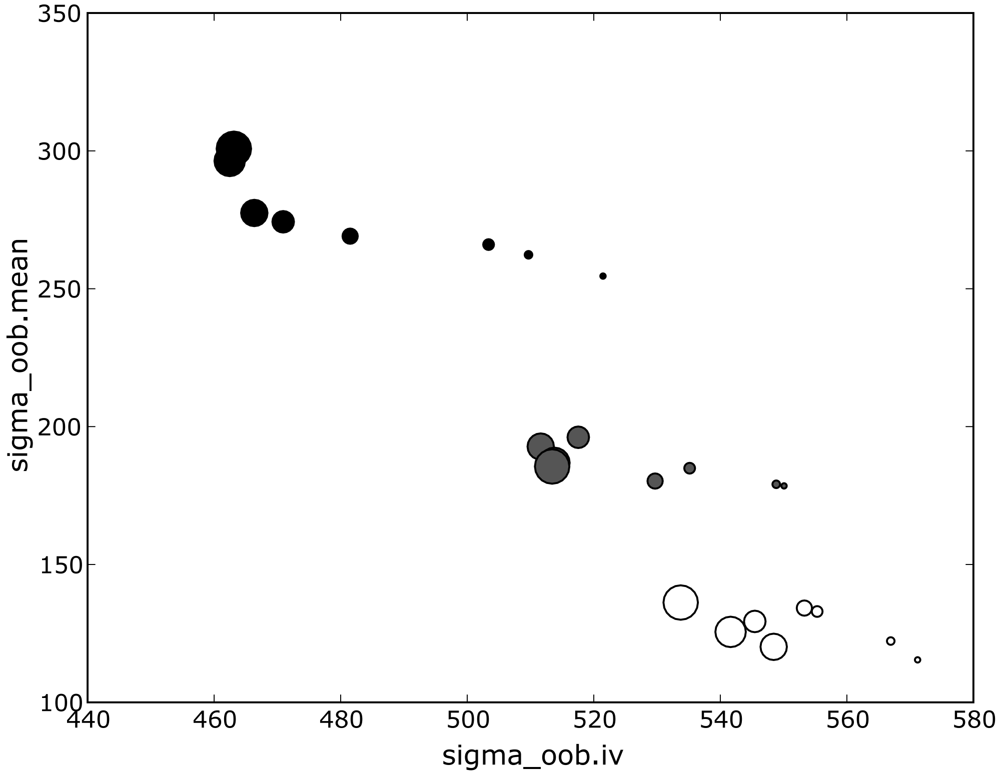

Figure 17: Black: $m_{leaf} = 250$, grey: $m_{leaf} = 500$, white: $m_{leaf} = 1000$. The size of a dot corresponds to $m_{try} \in \{2, 4, 8, 16, 32, 48, 64, 82\}$

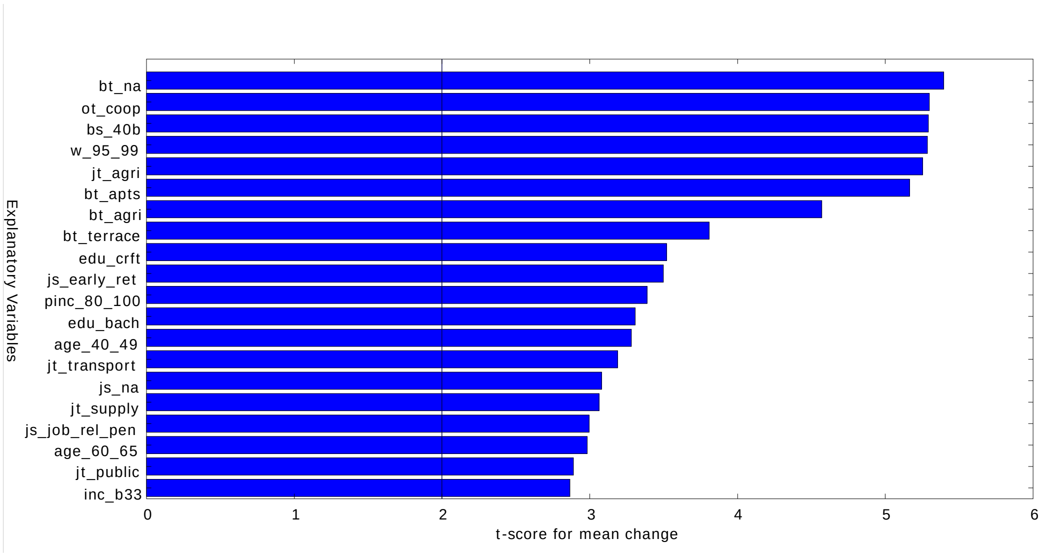

Figure 18: Top 20 variable importance. t–score is for mean change in pseudo chi-square fit measure

4.5 Variable Importance

The top 20 most important variables are shown in figure 18 p. 107 and the score for all variables are shown in figure 19 p. 108. Significant t-values are shown in bold.

The important measures are not very high, compared to the simulation case. This has to do with the multicollinearity in the data. As an example, if w_95_99 is very high it will almost always be the case that w_5_25 is very low. If two variables were perfectly correlated, both importance measures could be expected to be approximately halved.

We see that expected lifetime value and variance varies with the degree to which building types in an area is unknown, the degree to which households are cooperatives, if building sizes are below 40 square meters, the proportion of very wealthy households, the presence of jobs in agriculture, the proportion of apartments in a place, and so on. The next section will hand out more useful information on how lifetime varies, so no further attempt to interpret the patterns in importance will be carried out. However, there is one interesting observation on the importance measures within groups. For the building sizes, household wealth and personal income groups, it seems that the algorithm finds more information in the extremes than in the middle. For example, for building sizes, bs_40b and bs_130p are significant, while the remaining intervals in-between are not. This is probably often the case in a marketing setting, that data about extremes is more valuable than data about the average.

The importance measure could help an analyst eliminate some variables to simplify the model. In other scenarios we could have a much higher range of insignificant variables, but it is typical for sociodemographic data, that most attributes alone are weak predictors. One strength of a tree ensemble is the ability to cope with multicollinearity. Importance measures might be affected, but the predictive properties of an ensemble can be shown to be unaffected by the presence of the phenomena.

4.6 Segmentation

A two-component multidimensional scaling of the proximity matrix is show in figure 20 p. 109. There seems to be a gradual separation of high- to lower value agents, with

Map Coordinates	x	0.70			
	y	1.54			
	cap_dist	1.77			
Age	age_0_11	1.6	**Education**	edu_primary	1.33
	age_12_16	0.1		edu_gym	0.52
	age_17_22	1.5		edu_egym	1.37
	age_23_29	1.6		**edu_crft**	**3.52**
	age_30_39	1.0		edu_short	1.52
	age_40_49	**3.3**		**edu_medium**	**2.43**
	age_50_59	**2.0**		**edu_bach**	**3.31**
	age_60_65	**3.0**		**edu_grad**	**2.72**
	age_66p	**2.0**		edu_na	1.02
Job Status	**js_selfemp**	**2.06**	**Job Type**	**jt_agri**	**5.25**
	js_basic	**2.16**		**jt_manufact**	**2.59**
	js_mid	0.94		**jt_supply**	**3.06**
	js_hi	1.45		**jt_construct**	**2.32**
	js_topleader	1.04		jt_trade	1.78
	js_cashbenefit	1.65		**jt_transport**	**3.19**
	js_erl_ret	**3.50**		jt_finance	1.95
	js_j_rel_pen	**2.99**		**jt_public**	**2.89**
	js_student	0.29		jt_na	1.69
	js_pension	1.28		jt_child	0.68
	js_na	**3.08**		**jt_unempl**	**2.49**
	js_child	**2.62**		**jt_pension**	**2.35**
Building Types	**bt_agri**	**4.57**	**Ownership Types**	ot_owner	0.68
	bt_house	-0.33	(house ownership)	ot_privaterent	0.65
	bt_mult	0.60		ot_publicrent	0.43
	bt_terrace	**3.81**		**ot_coop**	**5.30**
	bt_apts	**5.16**			
	bt_na	**5.39**	**Building Sizes**	**bs_40b**	**5.29**
				bs_40_70	1.24
Family Type	**f_sic**	**2.83**		bs_71_90	1.71
	f_sinc	1.54		bs_91_130	0.46
	f_dic	1.53		**bs_130p**	**2.78**
	f_dinc	**2.50**			
	f_na	**2.78**	**Household Income**	**inc_b33**	**2.87**
				inc_33_75	**2.40**
Household Wealth	w_0_1	0.94		**inc_75p**	**2.22**
	w_1_5	**2.36**			
	w_5_25	**2.24**	**Personal Income of Highest Household Earner**	pinc_b20	1.36
	w_25_50	1.53		**pinc_20_40**	**2.10**
	w_50_75	0.48		pinc_40_60	1.64
	w_75_90	**2.41**		pinc_60_80	-0.03
	w_90_95	1.96		**pinc_80_100**	**3.39**
	w_95_99	**5.29**			
	w_99_100	**2.02**			

Figure 19: Variable importance. t-score for mean change in pseudo chi-square measure

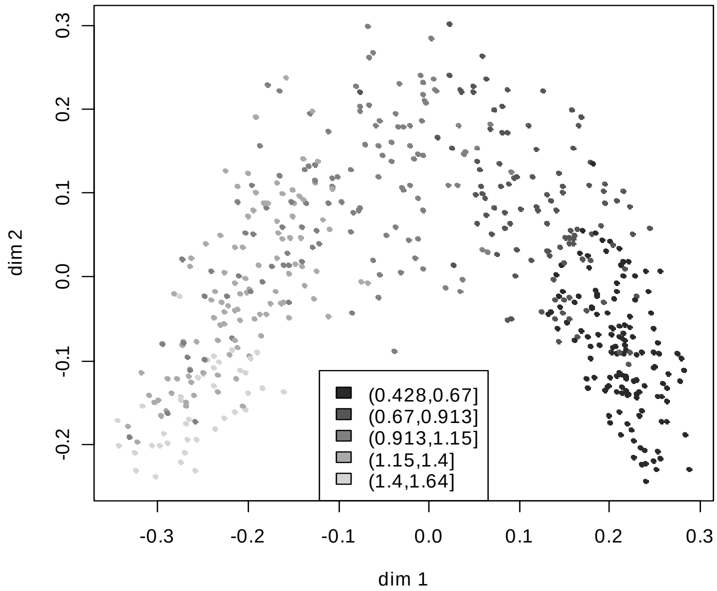

Figure 20: Plot of two dimensions from a multidimensional scaling of the proximity matrix. Colors represent an index of est. ltv over a full-group average ltv.

the extreme segments well-separated. We can use this information when thinking about forming segments. Focusing on the least valuable 20% (low group) and the most valuable 20% (high group), no distinct segments within these groups reveal themselves in the figure, making the task of characterizing each segment as an entity less hazardous.

A table of attribute means conditional on low group and high group membership is shown in figure 21 p. 110. Each group contains 2048 agents. Standard deviations are not reported, because at 2000+ observations, even tiny differences will be significant.

Scanning through the figure, a picture of low group members emerge. Young people are overrepresented, unmarried and living alone. They live in public or private rented apartments or coops of size 40-70 m^2 in cities. They have a primary school education or a craft behind them. Their income is somewhat below average and their wealth is considerably below average. An above average proportion is unemployed or otherwise involved in a public pension scheme.

The high group is to a large degree made up of mature couples in their 40's and 50's, with children having left the home or soon reaching that phase. The overwhelming majority live in their own house of 130 m^2 or more. University degrees are strongly over-represented along with medium length educations. Income and wealth are considerably above average.

Looking at importance scores of variables, it is interesting to note that the, say, top 5 most important variables are represented by extreme opposite means in the two segments. For example, the top 4 `w_99_100` is at `0.1` in the low group and at `5.5` in the high group, while the top 2 `ot_coop` is at `25.9` in the low group and comes out at `0.4` in the other group. These variables seperate the two extreme segments well.

		Low-group mean	High-group mean
Map Coordinates	x	7.14	7.07
	y	6.17	6.18
	cap_dist	153.0	265.0
Age	age_0_11	11.9	16.8
	age_12_16	3.5	6.9
	age_17_22	7.0	5.2
	age_23_29	18.9	3.3
	age_30_39	19.0	13.2
	age_40_49	11.1	16.1
	age_50_59	9.9	17.7
	age_60_65	4.8	8.6
	age_66p	13.8	12.2
Job Status	js_selfemp	2.1	5.9
	js_basic	32.5	25.7
	js_mid	7.7	11.0
	js_hi	6.4	11.9
	js_topleader	0.4	2.6
	js_cashbenefit	2.4	0.1
	js_erl_ret	5.8	1.4
	js_j_rel_pen	2.0	3.2
	js_student	3.8	2.7
	js_pension	12.9	9.1
	js_na	10.1	5.5
	js_child	13.8	20.8
Building Types	bt_agri	0.1	7.7
	bt_house	1.4	73.1
	bt_mult	6.9	11.5
	bt_terrace	1.2	2.2
	bt_apts	88.7	4.8
	bt_na	1.9	0.7
Family Type	f_sic	6.7	2.1
	f_sinc	56.8	16.4
	f_dic	10.1	32.0
	f_dinc	18.9	42.6
	f_na	7.4	6.9
Household Wealth	w_0_1	0.3	1.4
	w_1_5	3.0	2.8
	w_5_25	42.2	4.5
	w_25_50	29.6	5.9
	w_50_75	18.9	18.5
	w_75_90	4.5	26.1
	w_90_95	0.8	16.8
	w_95_99	0.5	18.5
	w_99_100	0.1	5.5

		Low-group mean	High-group mean
Education	edu_primary	26.8	8.6
	edu_gym	9.2	2.2
	edu_egym	2.4	0.8
	edu_crft	29.9	34.2
	edu_short	4.3	7.2
	edu_medium	9.9	21.1
	edu_bach	3.3	1.5
	edu_grad	7.2	21.2
	edu_na	7.1	3.2
Job Type	jt_agri	0.2	1.8
	jt_manufact	4.4	6.1
	jt_supply	0.2	0.4
	jt_construct	2.2	2.9
	jt_trade	9.5	9.8
	jt_transport	4.0	3.3
	jt_finance	9.6	12.0
	jt_public	18.9	20.5
	jt_na	0.2	0.4
	jt_child	15.5	21.5
	jt_unempl	14.7	7.6
	jt_pension	20.7	13.7
Ownership Types (house ownership)	ot_owner	8.7	93.3
	ot_privaterent	25.0	5.9
	ot_publicrent	40.4	0.3
	ot_coop	25.9	0.4
Building Sizes	bs_40b	5.3	0.3
	bs_40_70	51.0	1.6
	bs_71_90	27.7	5.3
	bs_91_130	14.0	27.8
	bs_130p	2.0	65.1
Household Income	inc_b33	43.2	7.4
	inc_33_75	41.1	23.2
	inc_75p	15.7	69.4
Personal Income of Highest Household Earner	pinc_b20	28.6	4.5
	pinc_20_40	24.9	8.7
	pinc_40_60	19.7	12.2
	pinc_60_80	16.4	21.3
	pinc_80_100	10.3	53.3

Figure 21: Attribute mean values for least valuable 20% of agents and for the most valuable 20%.

4.7 Segment Behavior

Continuing our exploration of the low and high group, the groups' estimated average lifetime value along with the initial distribution is shown in figure 22 on p. 112. It is a bit surprising to see only small differences in the initial state relative frequency. The low group is a bit more likely to enter through IDQ than the high group, while the high group is a little more likely to commit to six months instead of a quarter. The initial state distribution is probably not an important source in understanding the behavioral difference between the two groups.

The expected lifetime value of entry through each state, however, displays big differences between the two groups. As an example, low group state 0 agents are worth \$167, while high group state 0 agents are worth \$575.

To better understand differences in behavior, we turn to individual transition probabilities. Tables of top 5 transition states are shown in figure 23 p. 113 and 24 on p. 114. Behavior graphs are shown in figures 25 and 26 on p. 115 and p. 116.

Taking the most frequent intitial state, IDQ, as a point of departure we see important differences in behavior. The sequence {IDQ, FDQ.1, FDQ.2,FDQ.3,FDQ.4} happens with markedly different transition probabilities. The low group will survive the trip with probability $0.45 \times 0.29 \times 0.35 \times 0.84 = 0.038$. The high group will enter with probability $0.64 \times 0.44 \times 0.34 \times 0.81 = 0.078$, an important relative diffence that is reflected in the estimated LTV of \$282 and \$499 for the low and high group, respectively. It can also be noted, that the difference stems from behavior within the first few states. Those with low reservation prices are weeded out after a round of full-price bills.

Behavior from the initial state IWQ is not so different between the two groups, but still results in very different lifetime value estimates. It is important to note that agents can return from state 0, and high group customers do that much more often than low group agent. Each quarter, there is a 0.041 chance that a high group agent returns, while the same probability for a low group agent is only 0.020.

Another frequent initial state for high group agents are IDB, a six month introductory offer. High group agents go through a transition to FDQ.1 with probability 0.784, while low group customers get there with probability 0.568.

Some analysis could be applied to the non-introductory rebates, all R-states, but the sample is thin near those states. A bigger sample or a smaller relationship statespace is needed to solve that problem.

4.8 LTV and Prediction

In this section we will see if we can predict group differences in future revenue using out-of-bag estimated average lifetime value. Conditioning on state IDQ, the most frequent initial state, observed discounted revenue is calculated 24 months into the future. Agents with event histories below this thresold are excluded from the calculation. Note that this does not impose a selection bias, as event histories include state 0 events. $n = 1817$ agents are available for the calculation. The sample is partitioned into five segment of equal size, around 363 in each, according to estimated lifetime value. For each partition, the corresponding mean and standard deviation of the 24 months discounted revenue is calculated.

Figure 27 p. 117 displays a table of t-tests for differences in mean. It is peculiar that the second highest ranked segment actually scores the highest revenue, but the variance is also much higher and the difference is not significant. All in all the lifetime value seems to rank future revenue rather well.

	Low Group			High Group			
State	n	p	LTV	n	p	LTV	delta(LTV)
0	0	0.000	167	1	0.001	575	408
FDQ.1	40	0.023	308	45	0.030	805	497
FDQ.2	0	0.000	532	0	0.000	997	465
FDQ.3	0	0.000	963	0	0.000	1510	547
FDQ.4	0	0.000	1030	0	0.000	1612	582
FDY.1	4	0.002	1436	2	0.001	1610	174
FDY.2	0	0.000	1746	0	0.000	2321	576
FDY.3	0	0.000	1870	0	0.000	2449	579
FWQ.1	19	0.011	246	24	0.016	648	402
FWQ.2	0	0.000	280	0	0.000	670	390
FWQ.3	0	0.000	388	0	0.000	773	385
FWQ.4	0	0.000	400	0	0.000	781	381
FWY.1	0	0.000	324	2	0.001	723	399
FWY.2	0	0.000	506	0	0.000	859	353
FWY.3	0	0.000	638	0	0.000	980	341
IDB	126	0.072	310	196	0.130	813	503
IDM	68	0.039	216	63	0.042	622	406
IDQ	927	0.527	282	691	0.458	781	499
IDY	0	0.000	1317	46	0.030	1522	205
IWB	104	0.059	240	117	0.077	628	388
IWM	177	0.101	176	99	0.066	593	418
IWQ	279	0.159	238	184	0.122	640	402
IWY	16	0.009	346	40	0.026	761	414
RDQ	0	0.000	577	0	0.000	1029	452
RDY	0	0.000	1289	0	0.000	1621	332
RWQ	0	0.000	247	0	0.000	674	427
RWY	0	0.000	417	0	0.000	737	320
Sum	1760	1		1510	1		

Figure 22: Low and high group. Estimated average lifetime value and initial state count n and relative frequency p.

Low Group
Source state(n): (top_i_destination_state: probability)
0 (11526): (0 : 0.980) (IDQ : 0.005) (IWQ : 0.004) (IDB : 0.002) (IWB : 0.002)
FDQ.1 (600): (0 : 0.678) (FDQ.2 : 0.293) (RDQ : 0.013) (FWQ.2 : 0.010) (RWQ : 0.003)
FDQ.2 (158): (0 : 0.551) (FDQ.3 : 0.348) (RDQ : 0.051) (RDY : 0.019) (FWQ.3 : 0.019)
FDQ.3 (58): (FDQ.4 : 0.845) (0 : 0.138) (FWQ.4 : 0.017)
FDQ.4 (380): (FDQ.4 : 0.889) (0 : 0.082) (RDY : 0.013) (RDQ : 0.005) (FDY.2 : 0.005)
FDY.1 (32): (FDY.2 : 0.656) (0 : 0.125) (RDQ : 0.094) (RDY : 0.063) (FWY.2 : 0.031)
FDY.2 (21): (FDY.3 : 0.810) (0 : 0.143) (RDY : 0.048)
FDY.3 (28): (FDY.3 : 0.857) (0 : 0.071) (RDY : 0.071)
FWQ.1 (327): (FWQ.2 : 0.505) (0 : 0.483) (RDQ : 0.003) (RWQ : 0.003) (FDQ.2 : 0.003)
FWQ.2 (161): (0 : 0.559) (FWQ.3 : 0.366) (RWQ : 0.043) (RWY : 0.019) (FDQ.3 : 0.012)
FWQ.3 (62): (FWQ.4 : 0.823) (0 : 0.177)
FWQ.4 (249): (FWQ.4 : 0.843) (0 : 0.133) (RWQ : 0.020) (RDY : 0.004)
FWY.1 (15): (0 : 0.533) (RWY : 0.333) (FWY.2 : 0.133)
FWY.2 (6): (FWY.3 : 0.500) (0 : 0.333) (RWY : 0.167)
FWY.3 (4): (FWY.3 : 0.750) (0 : 0.250)
IDB (146): (FDQ.1 : 0.568) (0 : 0.308) (IDB : 0.110) (IDM : 0.007) (IDQ : 0.007)
IDM (76): (0 : 0.829) (FDQ.1 : 0.079) (IDQ : 0.066) (FWQ.1 : 0.013) (FDY.1 : 0.013)
IDQ (1027): (0 : 0.480) (FDQ.1 : 0.455) (IDQ : 0.055) (FWQ.1 : 0.006) (IDY : 0.002)
IDY (19): (FDY.1 : 0.684) (FDQ.1 : 0.105) (0 : 0.105) (IDY : 0.105)
IWB (124): (FWQ.1 : 0.645) (0 : 0.290) (IWB : 0.056) (IDQ : 0.008)
IWM (188): (0 : 0.878) (IWQ : 0.074) (FWQ.1 : 0.037) (IWM : 0.011)
IWQ (344): (FWQ.1 : 0.581) (0 : 0.360) (IWQ : 0.049) (IWM : 0.009)
IWY (24): (FWY.1 : 0.542) (0 : 0.250) (FWQ.1 : 0.125) (IWY : 0.083)
RDQ (40): (FDQ.1 : 0.475) (0 : 0.175) (RDY : 0.100) (RDQ : 0.075) (FDY.1 : 0.075)
RDY (14): (FDY.1 : 0.714) (FDQ.1 : 0.143) (0 : 0.143)
RWQ (6): (FWQ.1 : 0.500) (0 : 0.333) (RWQ : 0.167)
RWY (5): (FWY.1 : 0.600) (RWY : 0.200) (FWQ.1 : 0.200)

Figure 23: Low group. Top 5 transition probabilities from source state (first entry in row) to destination state (entry two and onwards).

High Group
Source state(n): (top_i_destination_state: probability)
0 (9313): (0 : 0.959) (IDY : 0.010) (IDB : 0.008) (IDQ : 0.006) (IWB : 0.006)
FDQ.1 (769): (0 : 0.525) (FDQ.2 : 0.442) (RDQ : 0.016) (FWQ.2 : 0.010) (RDY : 0.004)
FDQ.2 (297): (0 : 0.525) (FDQ.3 : 0.340) (RDQ : 0.081) (RDY : 0.037) (FWQ.3 : 0.007)
FDQ.3 (98): (FDQ.4 : 0.806) (0 : 0.163) (FDY.2 : 0.020) (RDY : 0.010)
FDQ.4 (1095): (FDQ.4 : 0.922) (0 : 0.058) (FDY.2 : 0.010) (RDY : 0.005) (RDQ : 0.003)
FDY.1 (119): (FDY.2 : 0.387) (RDY : 0.286) (0 : 0.286) (RDQ : 0.034) (FDQ.4 : 0.008)
FDY.2 (64): (FDY.3 : 0.828) (0 : 0.078) (RDY : 0.063) (FDQ.4 : 0.031)
FDY.3 (153): (FDY.3 : 0.935) (0 : 0.059) (RDY : 0.007)
FWQ.1 (291): (FWQ.2 : 0.601) (0 : 0.381) (RWQ : 0.007) (RDQ : 0.003) (FWY.2 : 0.003)
FWQ.2 (168): (0 : 0.524) (FWQ.3 : 0.411) (RWQ : 0.054) (FWY.2 : 0.012)
FWQ.3 (70): (FWQ.4 : 0.857) (0 : 0.129) (RWQ : 0.014)
FWQ.4 (385): (FWQ.4 : 0.865) (0 : 0.117) (FWY.2 : 0.008) (RWQ : 0.005) (FDQ.4 : 0.003)
FWY.1 (34): (0 : 0.412) (FWY.2 : 0.382) (RWY : 0.176) (RWQ : 0.029)
FWY.2 (10): (FWY.3 : 0.600) (0 : 0.300) (RWY : 0.100)
FWY.3 (6): (FWY.3 : 0.833) (0 : 0.167)
IDB (245): (FDQ.1 : 0.784) (0 : 0.151) (IDB : 0.049) (IWB : 0.012) (FDY.1 : 0.004)
IDM (71): (0 : 0.873) (FDQ.1 : 0.070) (IDQ : 0.042) (FDY.1 : 0.014)
IDQ (769): (FDQ.1 : 0.637) (0 : 0.303) (IDQ : 0.051) (IDY : 0.003) (IDM : 0.003)
IDY (116): (FDY.1 : 0.698) (0 : 0.129) (FDQ.1 : 0.078) (IDY : 0.078) (IDM : 0.009)
IWB (140): (FWQ.1 : 0.700) (0 : 0.236) (IWB : 0.064)
IWM (125): (0 : 0.808) (IWQ : 0.104) (FWQ.1 : 0.064) (IDY : 0.008) (IWM : 0.008)
IWQ (225): (FWQ.1 : 0.640) (0 : 0.258) (IWQ : 0.053) (IWM : 0.044) (IWY : 0.004)
IWY (64): (FWY.1 : 0.656) (0 : 0.172) (IWY : 0.094) (IDY : 0.031) (FWQ.1 : 0.031)
RDQ (46): (FDQ.1 : 0.522) (RDY : 0.174) (0 : 0.109) (FWQ.1 : 0.087) (RDQ : 0.043)
RDY (52): (FDY.1 : 0.750) (FDQ.1 : 0.192) (RDY : 0.058)
RWQ (8): (FWQ.1 : 1.000)
RWY (4): (FWQ.1 : 0.500) (FWY.1 : 0.500)

Figure 24: High group. Top 5 transition probabilities from source state (first entry in row) to destination state (entry two and onwards).

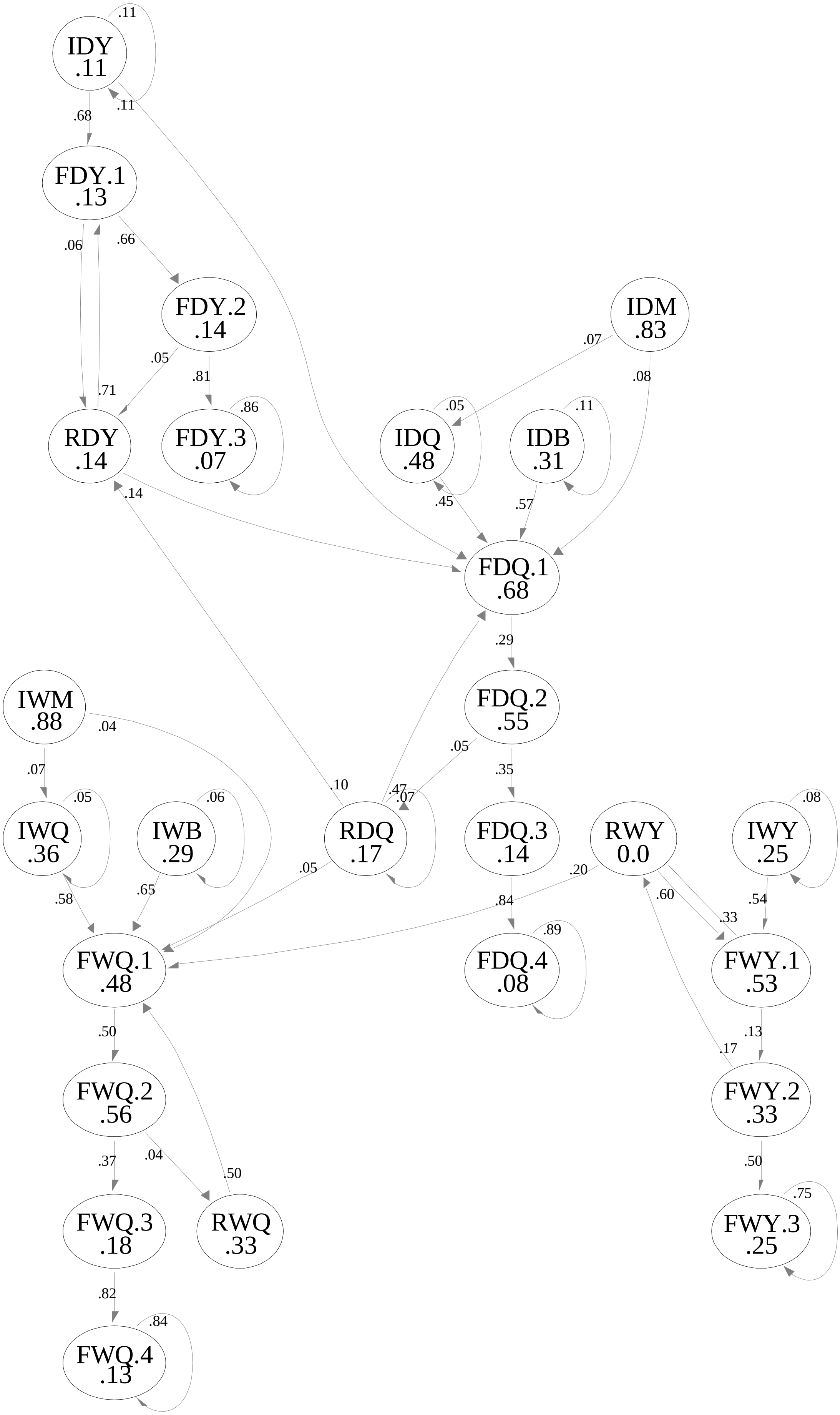

Figure 25: Low group. Behavior graph.

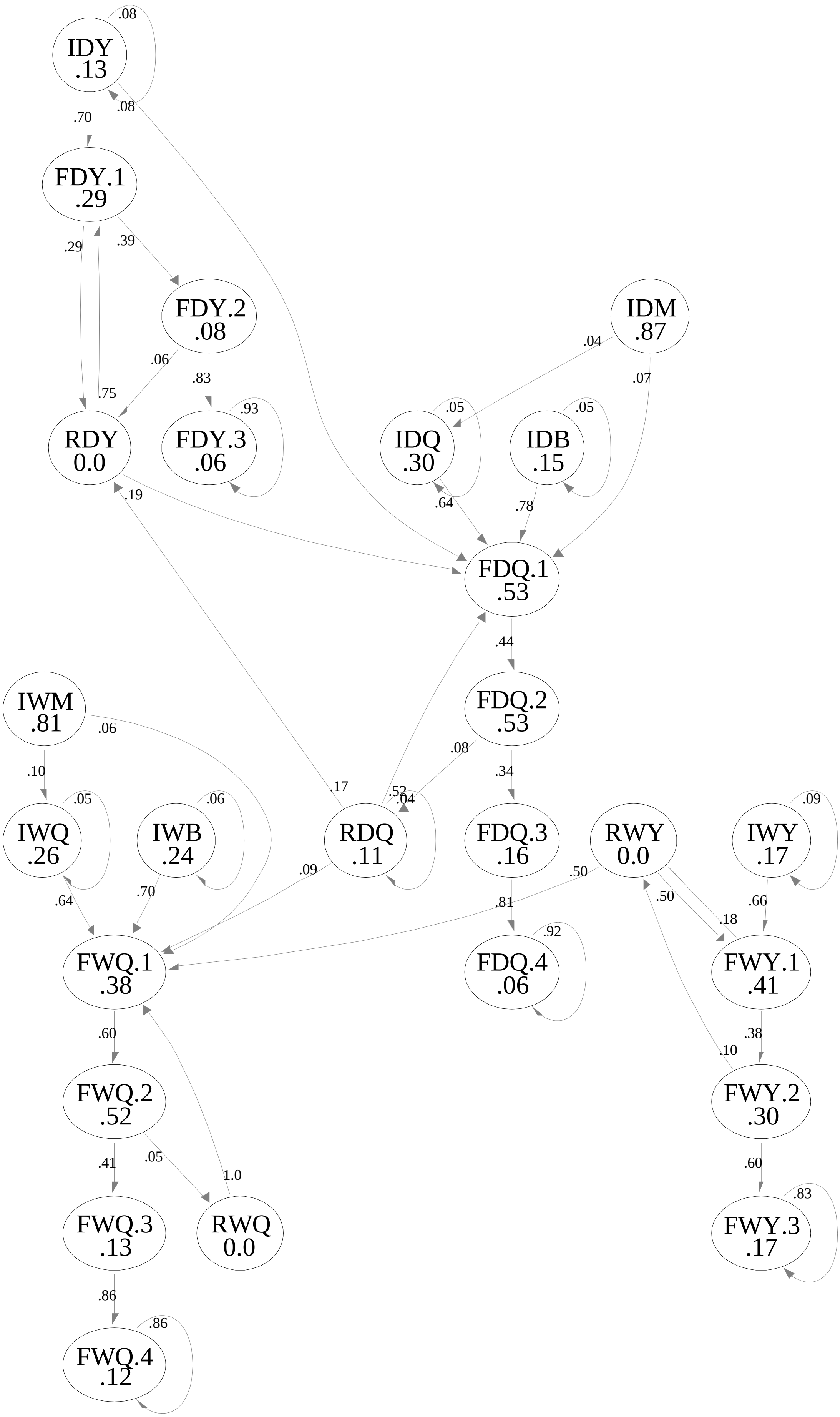

Figure 26: High group. Behavior graph.

	Estimated LTV [255,321]	(321,382]	(382,524]	(524,653]	(653,907]
Mean	44.4	49.8	67.8	78.3	76.9
SD	63.7	66.2	76.5	98.7	67.8
Estimated LTV					
[255,321]	0.0	1.1	**4.5**	**5.5**	**6.6**
(321,382]	1.1	0.0	**3.4**	**4.6**	**5.5**
(382,524]	**4.5**	**3.4**	0.0	1.6	1.7
(524,653]	**5.5**	**4.6**	1.6	0.0	0.2
(653,907]	**6.6**	**5.5**	1.7	0.2	0.0
n=	1817				

Figure 27: Mean, standard deviation and t-tests for difference in mean for 24 months of discounted revenue, grouped by estimated lifetime value.

4.9 Perspective to Other Models

It is difficult to compare the presented approach to other models. The outcome variable is a complicated function of a matrix, not a simple univariate outcome. This makes direct comparisons with methods such as logistics regression and simple duration models impossible. The recency, frequency and monetary value approach could be compared to a version of the model focused on predicting customer value from the last observed path, but the application section in this paper takes the initial state as the point of departure. Directed clustering methods also need a univariate outcome variable, while non-directed clustering methods do not. The latter set of methods could be used to form segments for comparison, but then a comparison criteria would be needed. Predicted revenue is one such measure, as used in the previous section, but it is important to note that the tree ensemble does not optimize for prediction. The problem can in some respect be compared to the challenge involved in comparing any segmentation method, that it is difficult to claim that one set of segmentation structures are better than another set. When arguing for and against the approach presented in this paper, the author will claim that optimizing directly for lifetime value has a high degree of face value in the context of business decision making..

5 Conclusion

The presented model seems like a useful tool for anyone exploring event histories with some type of revenue or reward involved. Many of the attractive properties of decision trees are kept with tree ensembles, while a range of useful additional information can be learned, such as variable importance and case proximity. The application of the model resulted in a meaningful segmentation of current and former customers, yielding easily interpretable results. A simple test of predictive performance did also yield a positive result. Despite a lack of knowledge regarding aspects of the model's statistical properties, simulation and application results provide encouraging evidence for the viability of the approach.

References

Breiman, L. (2001), 'Random forests', Machine Learning 45(1), 5–32.

Breiman, L. & Cutler, A. (2001), 'Random forests (program code)'.

Buckinx, W. & Van den Poel, D. (2005), 'Customer base analysis: partial defection of behaviourally loyal clients in a non-contractual fmcg retail setting', European Journal of Operational Research 164(1), 252–268.

Cox, T. & Cox, M. A. A. (1994), Multidimensional Scaling, Chapman and Hall.

Hamza, M. & Larocque, D. (2005), 'An empirical comparison of ensemble methods based on classification trees', Journal of Statistical Computation and Simulation 75(8), 629–643.

Lariviere, B. & Poel, D. V. d. (2005), 'Predicting customer retention and profitability by using random forests and regression forests techniques', Expert Systems with Applications 29(2), 472–484.

Liaw, A. & Wiener, M. (2002), 'Classification and regression by randomforest', R News 2(3), 18–22.

Sobel, M. J. (1982), 'The variance of discounted markov decision processes', Journal of Applied Probability 19(4), 794–802.

Working Papers published by the Department of Management

2005-1: Esben Koling Laustrup and Johannes Raaballe: Udbytteannonceringseffekten i Danmark.

2005-2: Carina Sponholtz: Separating the Stock Market's Reaction to Simultaneous Dividend and Earnings Announcements.

2005-3: Kyle Bagwell and Per Baltzer Overgaard: Look How Little I'm Advertising!

2005-4: Jerker Nilsson and Ole Øhlenschlæger Madsen: Cross-border Mergers Between Agricultural Co-operatives - a Governance Perspective.

2005-5: Bent Jesper Christensen and Morten Ørregaard Nielsen: The Effect of Long Memory in Volatility on Stock Market Fluctuations.

2005-6: Bent J. Christensen and Nicholas M. Kiefer: Investment in Advertising Campaigns and Search: Identification and Inference in Marketing and Dynamic Programming Models.

2005-7: Flemming Witt and Jørn Flohr Nielsen: The Difficult Empowerment in Danish Hospitals: Power to the Nurses!?

2006-1: Thomas Busch, Bent Jesper Christensen and Morten Ørregaard Nielsen: The Information Content of Treasury Bond Options Concerning Future Volatility and Price Jumps.

2006-2: Thomas Busch, Bent Jesper Christensen and Morten Ørregaard Nielsen: Forecasting Exchange Rate Volatility in the Presence of Jumps.

2006-3: Thomas Busch, Bent Jesper Christensen and Morten Ørregaard Nielsen: The Role of Implied Volatility in Forecasting Future Realized Volatility and Jumps in Foreign Exchange, Stock, and Bond Markets.

2006-4: Niels Stender: Expected Present Value of a Customer.

2006-5: Niels Stender: A Quantitative Model of Dynamic Customer Relationships.

2006-6: Niels Stender: Fast Kernel Regression.

2006-7: Niels Stender: Analyzing Customer Lifetime Value using Tree Ensembles.