

Churn Modeling for Mobile Telecommunications:

Winning the Duke/NCR Teradata Center for CRM Competition

N. Scott Cardell, Mikhail Golovnya, Dan Steinberg

Salford Systems

<http://www.salford-systems.com>

June 2003

The Churn Business Problem

- Churn represents the loss of an existing customer to a competitor
- A prevalent problem in retail:
 - Mobile phone services
 - Home mortgage refinance
 - Credit card
- Churn is a problem for any provider of a subscription service or recurring purchasable.
 - Costs of customer acquisition and win-back can be high
 - Much cheaper to invest in customer retention
 - Difficult to recoup costs of customer acquisition unless customer is retained for a minimum length of time
- Churn is especially important to mobile phone service providers
 - easy for a subscriber to switch services.
 - Phone number portability will remove last important obstacle

Churn a Core CRM issue

- **The core CRM issues include:**
 - **Customer acquisition**
 - **Customer retention**
 - **Cross-sell/Up Sell**
 - **Maximizing Lifetime Customer Value**
- **Churn can be combated by**
 - **Acquiring more loyal customers initially**
 - **Taking preventative measures with existing customers**
 - **Identifying customers most likely to defect**
 - **offering incentives to those customers you want to keep**
- **All CRM management needs to take churn into account**

Predicting Churn: Key to a Protective Strategy

- **Predictive modeling can assist churn management**
 - by tagging customers most likely to churn
- **High risk customers should first be sorted by profitability**
- **Campaign targeted to the most profitable at-risk customers**
 - **Typical retention campaigns include:**
 - Incentives such as price breaks
 - Special services available only to select customers
- **To be cost effective retention campaigns must be targeted to the right customers**
 - **Customers who would probably leave without the incentive**
 - **Costly to offer incentives to those who would stay regardless**

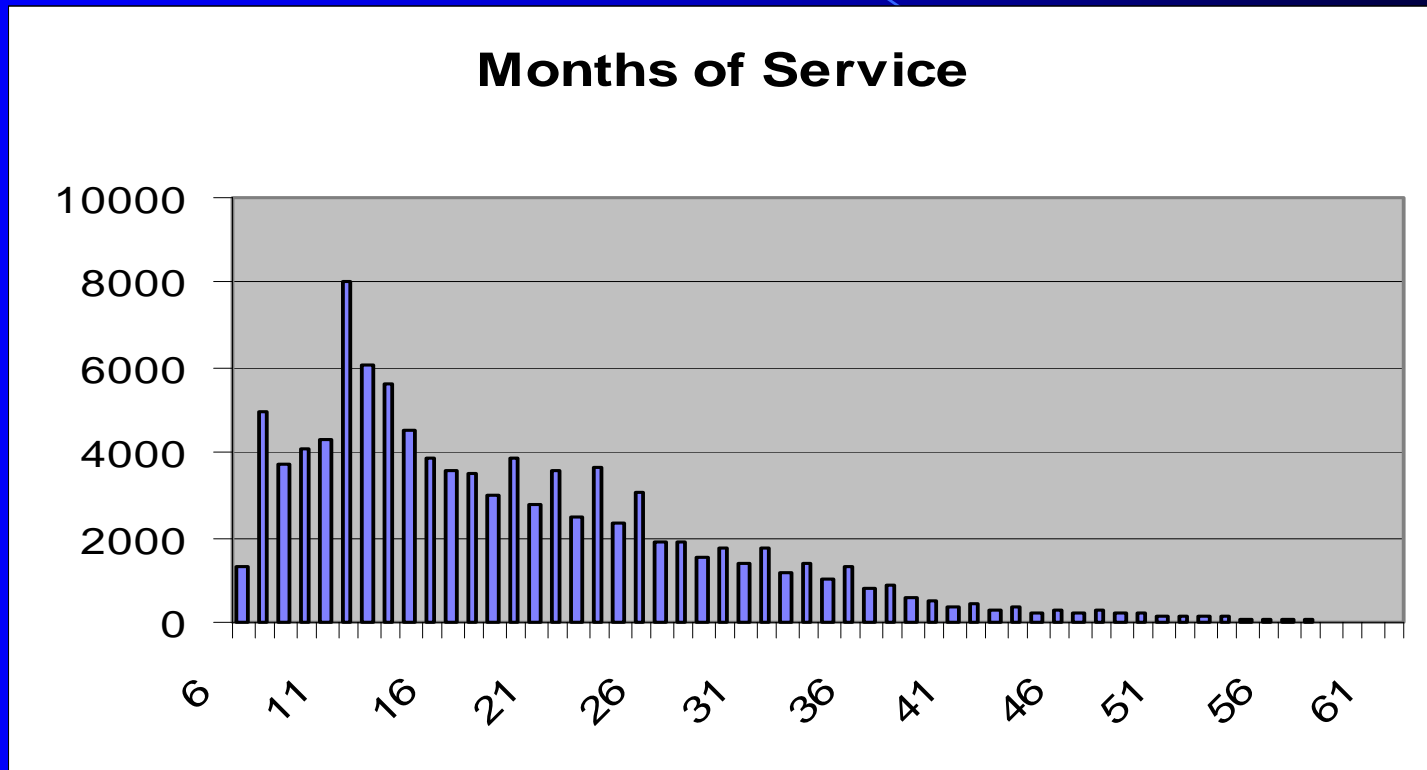
Duke/NCR Teradata 2003 Tournament

- The CRM Center sought to identify *best practice for churn modeling* in a real world context
- Solicited a major wireless telco to provide customer level data for an international modeling competition.
- Data suitable for churn modeling and prediction
- Competition was opened Aug 1, 2002 to all interested participants.
 - Publicized in a variety of data mining web sites, mailing lists, and SIGs (special interest groups)
 - Participants were given until January 10, 2003 to submit their predictions

Nature of the Data and Challenge

- **Data were provided for 100,000 customers with at least 6 months of service history**
 - One summary record per mobile phone account
 - Stratified into equal numbers of churners and non-churners
- **Historical information provided in the form of**
 - Type and price of current handset
 - Date of last handset change/upgrade
 - Total revenue (expenditure)
 - Broken down into recurring charges and non-recurring charges
 - Call behavior statistics (type, number, duration, totals, etc.)
 - Demographic and geographical information,
 - including familiar Acxiom style direct mail and marketing variables
 - Census-derived neighborhood summaries.

Accounts by Months of Service



Time Structure of Data

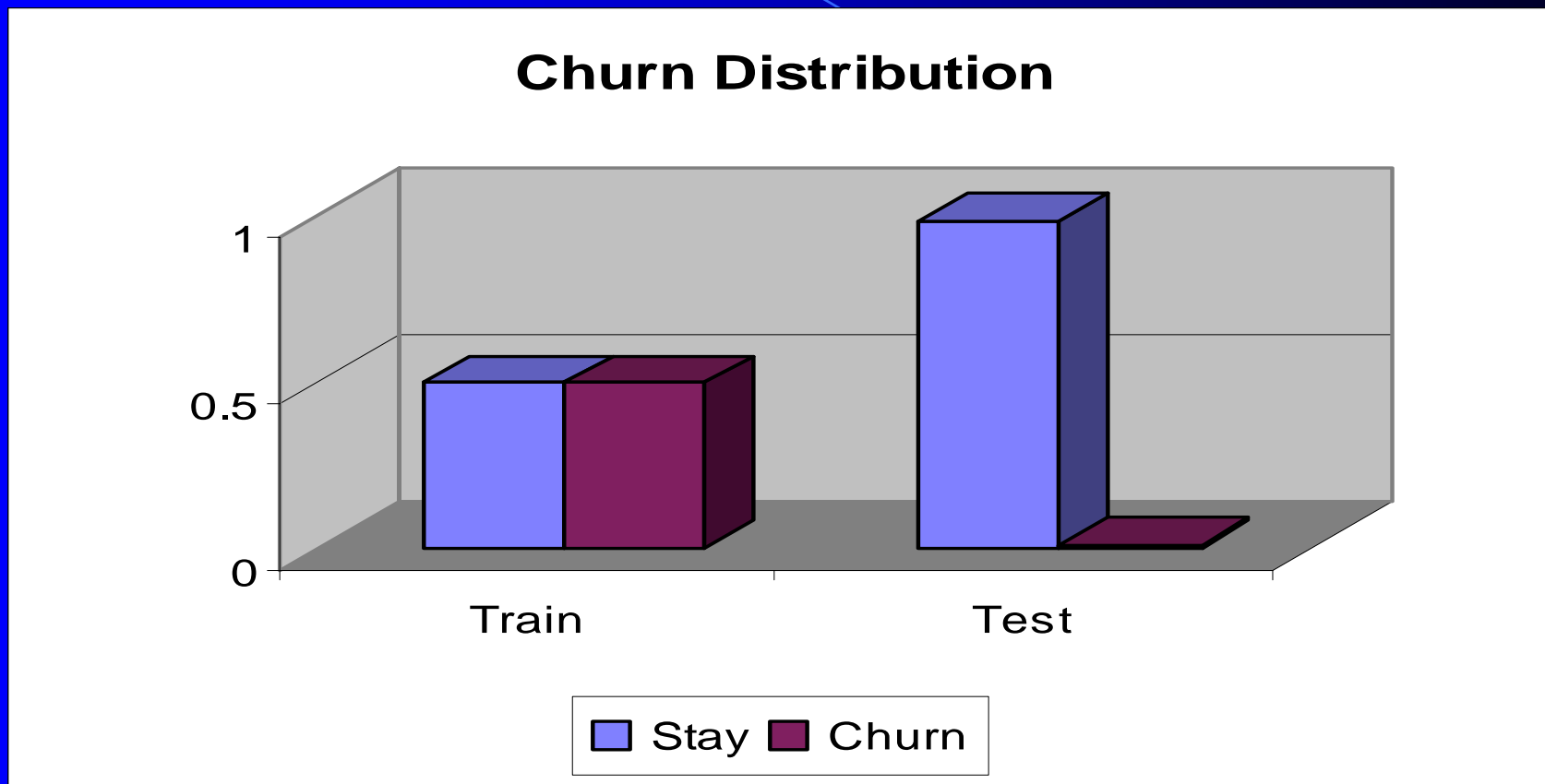
- **Data in the form of a “snapshot” of customer at a specific point in time**
 - Month of data capture was one of:
 - July, September, November, December, 2001
- **Historical data referred to**
 - Current period (current plus prior 3 months)
 - Prior 3 months, prior 6 months
 - Lifetime (at least 6 months, as much as 5 years)
- **Forecast Period**
 - Churn or not in period 31-60 days after “snapshot”

Time Shape of the Data

| | | | | | | | | Current Month | Forecast |
|--|--|--|--|--|--|--|--|-------------------|----------|
| | | | | | | | | Sample Point | Month |
| | | | | | | | | | |
| | | | | | | | | 3 month Averages | |
| | | | | | | | | 6 month Averages | |
| | | | | | | | | | |
| | | | | | | | | Lifetime Averages | |

- Data were captured for an account at a specific point in time
- From the perspective of that time retrospective summaries were computed
- Churn models were evaluated on forecast accuracy for the period 31 - 60 days
- To reflect data from different calendar months accounts were captured in July, September, November, December, 2001
- Which month a record was captured in was not available to the modelers

Train vs. "Test" Churn Distribution



Care required when forecasting from Train to Test data

Nature of Call Behavior Data

- **Summary statistics describing number, duration, etc. of:**
 - completed calls
 - failed calls
 - voice calls
 - data calls
 - call forwarding
 - customer care calls
 - directory info
- **Statistics included mean and range for**
 - Current period
 - Preceding 3 months,
 - Preceding 6 months
 - lifetime.

Evaluation Data

- **Models were evaluated by performance on two different groups of accounts**
 - **“Current” data**
 - Unseen accounts also drawn from July thru December 2001
 - **“Future” data**
 - Unseen accounts drawn from first half of 2002
- **Evaluation on “current” data is the norm**
 - Arranged by holding back some data for this purpose
- **“Future” data is a more realistic and stringent test**
 - Data used to test comes from a later time period
 - Markets, processes, behaviors change over time
 - Models tend to degrade over time due to changes in customer base and changes in offers by competitors, technology, etc.
- **Model for best future performance may not be best for current performance, and vice versa**

Modeling Observations

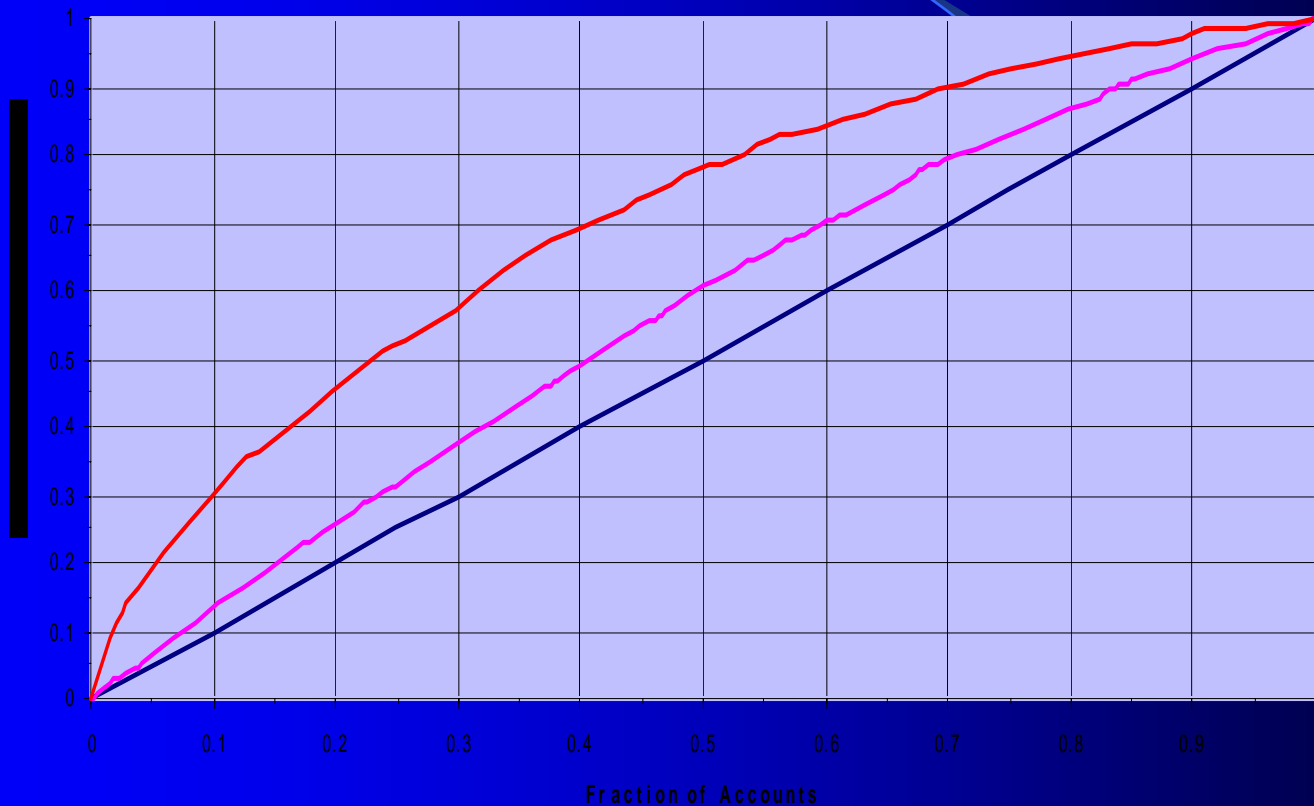
- **Competition defined a sharply defined task:**
 - churn within a specific window for existing customers of a minimum duration.
 - Objective was to predict probability of loss of a customer 30-60 days into the future.
- **Challenge was defined in a way to avoid complications of censoring**
 - Censored data could require survival analysis models.
- **Each customer history was already summarized.**
 - Only a modest amount of data prep required
 - No access to the raw data was provided so new summary construction was not possible
- **Data quality was good**
 - Perhaps because derived largely from operational database
- **Majority of effort could be devoted to modeling**

Model Evaluation

- **Lift in top decile**
 - Fraction of churners actually captured among the 10% “most likely to churn” as rated by the model
- **Overall model performance as measured by the Gini coefficient**
 - Area under the gains curve as illustrated on following slides
- **Measures calculated for two different time periods**
 - “Current” (June thru December 2001)
 - “Future” (First quarter 2002)

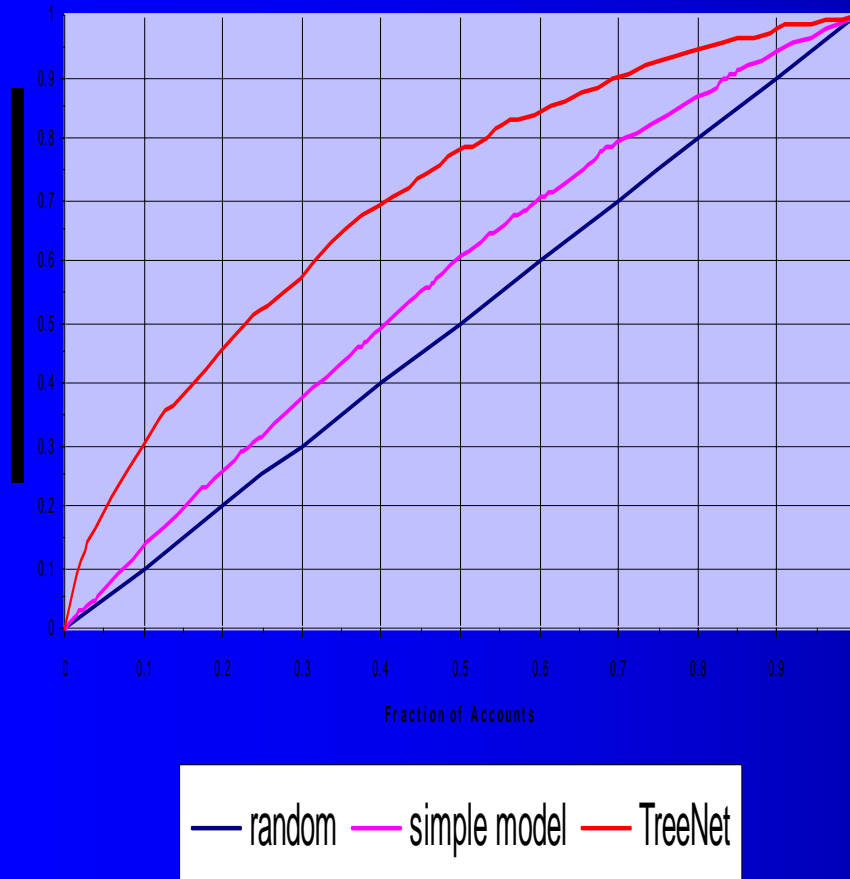
Two measures calculated on each of two time periods yields 4 performance indicators in total
- **Salford models were best in all four categories**

Gains Chart: TreeNet vs Other



— random — simple model — TreeNet

Top Decile Lift and Gini



- To produce the gains chart we order the data by the probability of event (churn)
- We plot the %target class captured as we move deeper into the ordered file
- In the example at left we capture 30% of churners in the top 10% of accounts and 70% in top 40% of accounts
- Models can also be evaluated by the “area under the curve” or Gini coefficient

Comparative Model Results: Top Decile Lift

| | Future Data | Current data |
|----------------------------------|----------------|---------------|
| Number of Accounts | 100,000 | 51,036 |
| Number Churning | 1,808 | 924 |
| Top decile capture | | |
| Salford entry | 525 | 278 |
| 2nd best model | 506 | 253 |
| Average of models | 387 | 193 |
| Salford Advantage | | |
| Over runner up | 3.8% | 9.9% |
| Over average | 35.7% | 44.0% |

Comparative Model Results: Overall Model Performance (Gini)

| | Future Data | Current data |
|----------------------------------|----------------|---------------|
| Number of Accounts | 100,000 | 51,036 |
| Number Churning | 1,808 | 924 |
| Gini Coefficient | | |
| Salford entry | .409 | .400 |
| 2nd best model | .370 | .361 |
| Average of models | .269 | .261 |
| Salford Advantage | | |
| Over runner up | 10.5% | 10.8% |
| Over average | 52.0% | 53.2% |

Comparative Model Results

| <u>Data Set</u> | <u>Measure</u> | <u>TreeNet Ensemble</u> | <u>Single TreeNet</u> | <u>2nd Best</u> | <u>Avg. (Std)</u> |
|-----------------|-----------------|-------------------------|-----------------------|----------------------------|-------------------|
| Current | Top Decile Lift | 2.90 | 2.88 | 2.80 | 2.14 (.536) |
| Current | Gini | .409 | .403 | .370 | .269 (.096) |
| Future | Top Decile Lift | 3.01 | 2.99 | 2.74 | 2.09 (.585) |
| Future | Gini | .400 | .403 | .361 | .261 (.098) |

Model Observations

- **Single TreeNet model always better than 2nd best entry in field.**
- **Ensemble of TreeNets slightly better than a single TreeNet 3 out of 4 times.**
- **TreeNet entries substantially better than the average.**
- **Minimal benefits from data preprocessing above and beyond that already embodied in the account summaries**
- **Virtually no manual, judgmental, or model guided variable selection**
 - **We let TreeNet do all the work of variable selection**

Business Benefits of TreeNet Model

- In broad telecommunications markets the added accuracy and lift of TreeNet should yield substantially increased revenue
- For each 5 million customers over a one year period our models could capture as many as 20,000 more churn accounts in top decile
- Average revenue per month per account is \$58.69 and \$704 per year
- A customer retention rate of 15% should yield over \$2 million per year in added revenue from the top decile alone
- The larger mobile telcos in the US and Europe have huge customer bases. Sprint PCS boasts 50 million accounts, so for Sprint the benefits could show in the vicinity of \$20 million per year in added revenue.

Data Preparation

- A minimal amount of data preprocessing was undertaken to repair and extend original data.
- Some missing values could be recoded to “0.”
- Select non-missing values were recoded to missing.
- Experiments with missing value handling were conducted, including the addition of missing value indicators to the data.
 - CART imputation
 - “All missings together” strategies in decision trees
 - Missings in a separate node
 - Missings go with non-missing high values
 - Missings go with non-missing low values

Modeling Tool of Choice: TreeNet™ Stochastic Gradient Boosting

- **TreeNet was key to winning the tournament.**
 - Provided much greater accuracy and top decile lift than any other modeling method we tried.
- **A new technology, different than standard boosting, developed by Stanford University Professor Jerome Friedman.**
- **Based on the CART® decision tree and thus inherits these characteristics:**
 - Automatic feature selection
 - Invariant with respect to order-preserving transforms of predictors
 - Immune to outliers
 - Built-in methods for handling missing values

How TreeNet Works

- Goal is to model a target variable Y as a function of the data X

- $$Y = F(X)$$

- We make this nonparametric by expressing the function as a series expansion

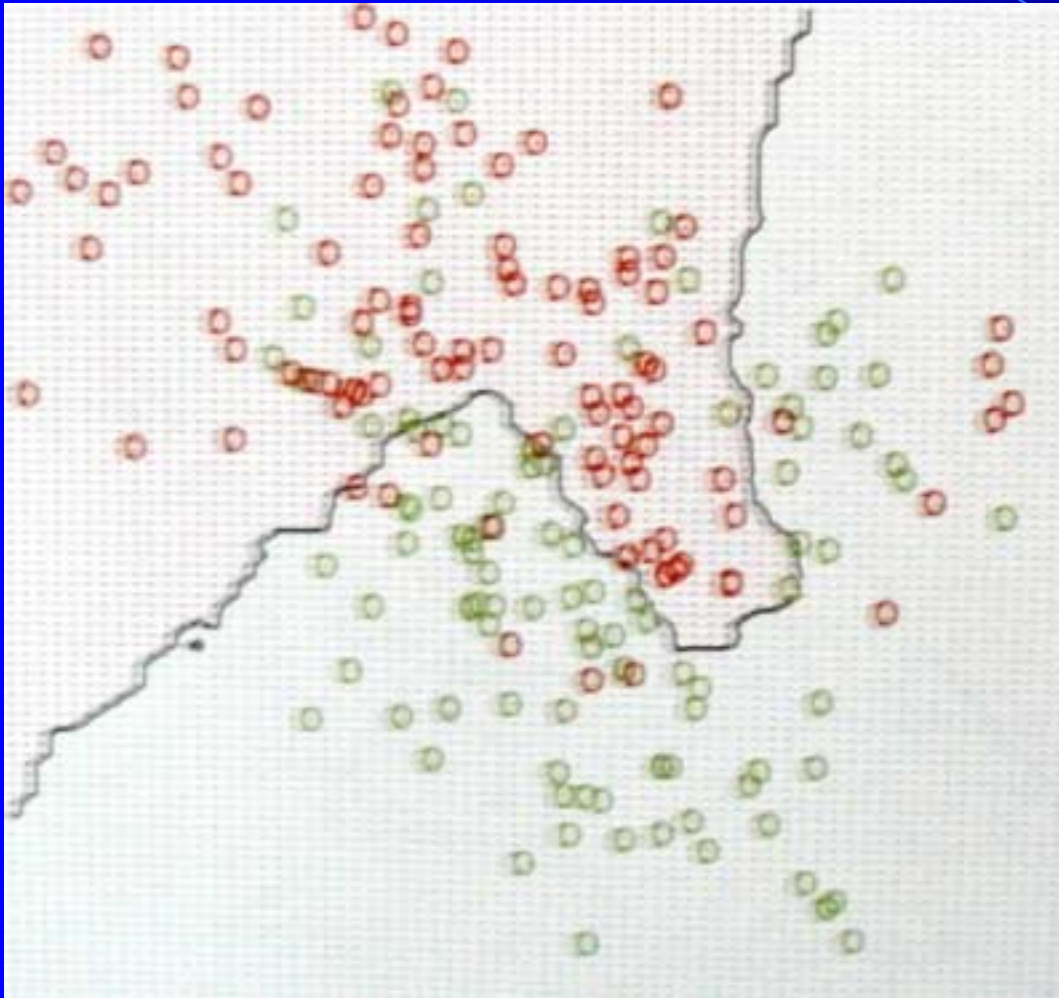
$$F(X) = F_0 + \beta_1 T_1(X) + \beta_2 T_2(X) + \dots + \beta_M T_M(X)$$

- Expansion is developed one stage at a time
 - Each stage is a new learning cycle starting again using “all” the data
 - A term is learned once and never updated thereafter
- Each term of the series will typically be a small decision tree
 - As few as 2 nodes, typically 4 to 6 nodes, occasionally more
- Fit obtained by optimizing an objective function
 - e.g: likelihood function or sum of squared errors.

TreeNet Mechanics

- Stagewise function approximation in which each stage models transformed target (e.g. residuals) from last step model
 - Each stage uses a very small tree, as small as 2 nodes and typically in the range of 4-9 nodes
 - Each stage is intended to learn only a little
- Each stage learns from a fraction of the available training data, typically less than 50% to start
 - Slow learning intended to protect against overfitting
 - Data fraction used often falls to 20% or less by the last stag
- Each stage updates model only a little: severely downweighted contribution of each new tree (learning rate is typically 0.10, even 0.01 or less)
- In classification, focus is on points near decision boundary; ignores points far away from boundary even if the points are on the wrong side

Decision Boundary: 2 predictors



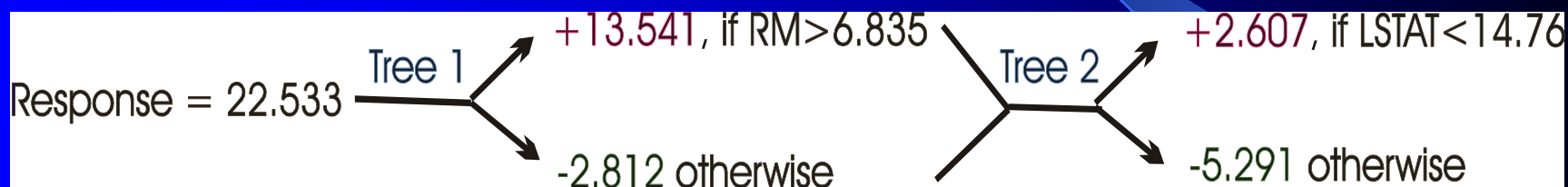
- Red dots represent YES (+1)
- Green dots represent NO (-1)
- Black curve is the current stage decision boundary
- TreeNet will not use data points too far from boundary to learn model update

TreeNet Objective Functions

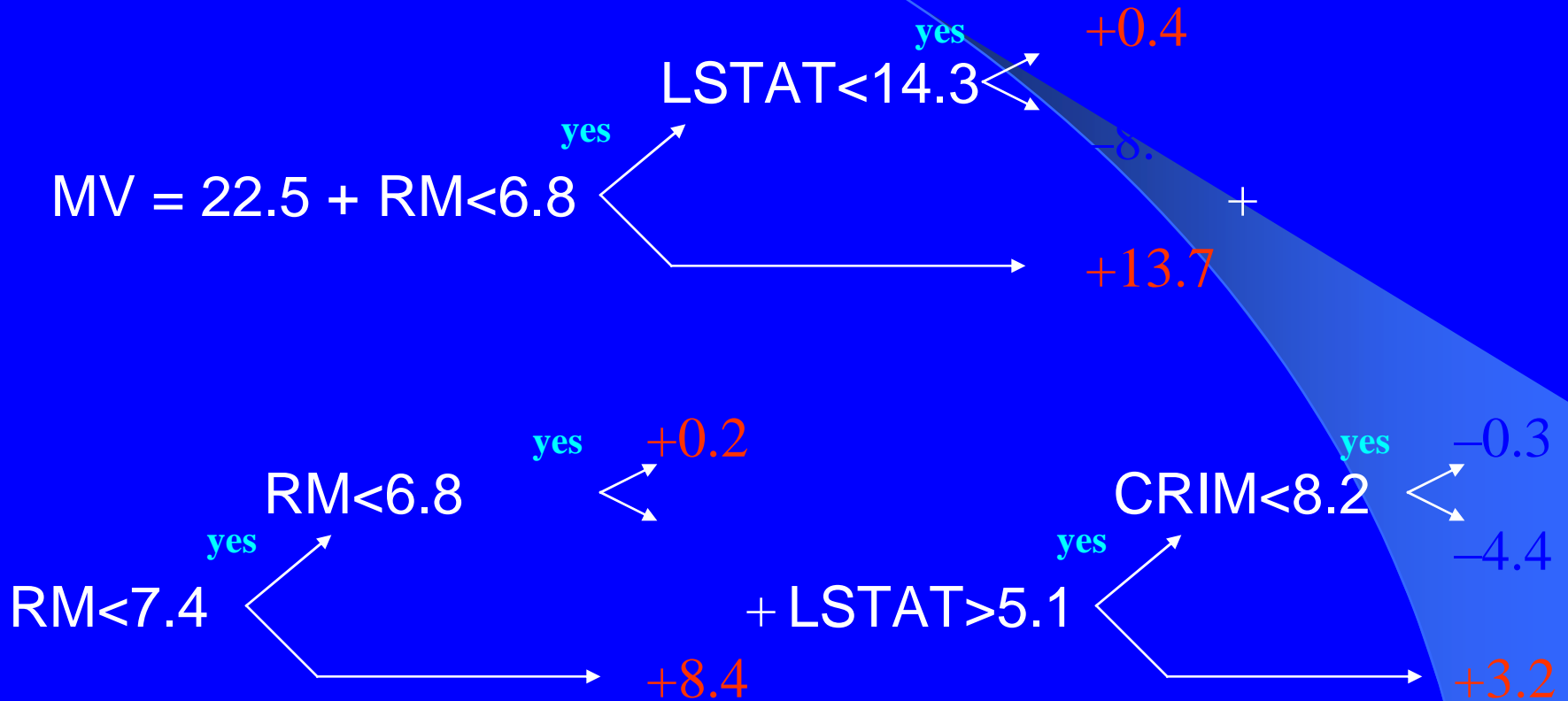
- **For categorical targets (classification)**
 - binary classification
 - multinomial classification
 - Logistic regression
- **For continuous targets (regression)**
 - least-squares regression
 - least-absolute-deviation regression
 - M-regression (Huber loss function)
- **Other objective functions are possible and will be added in the future**

Simple TreeNet Example

- First two stages of a regression model
 - Each stage below is a 2 node tree



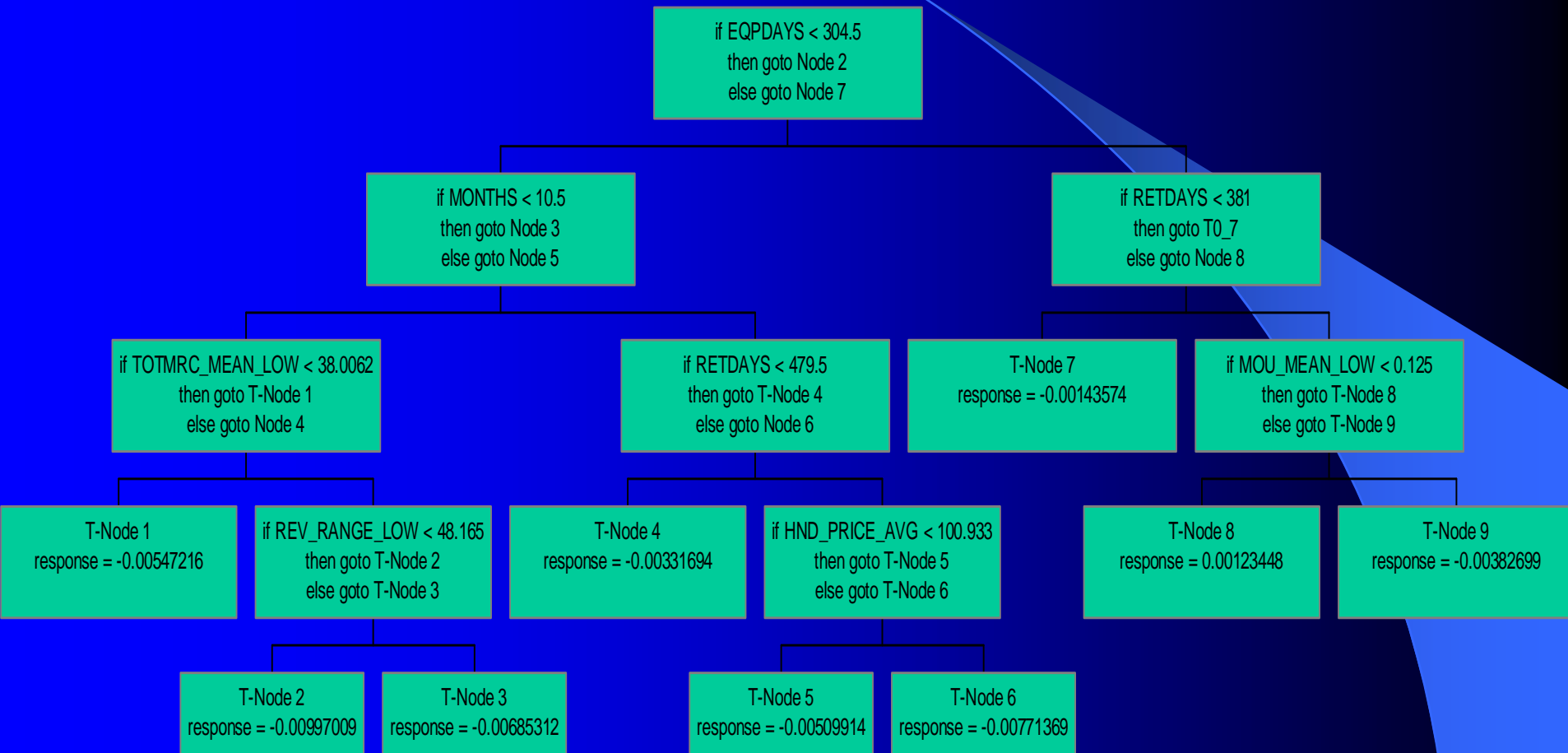
TreeNet Model with Three-Node Trees



Each tree has three terminal nodes, thus partitioning data at each stage into three segments

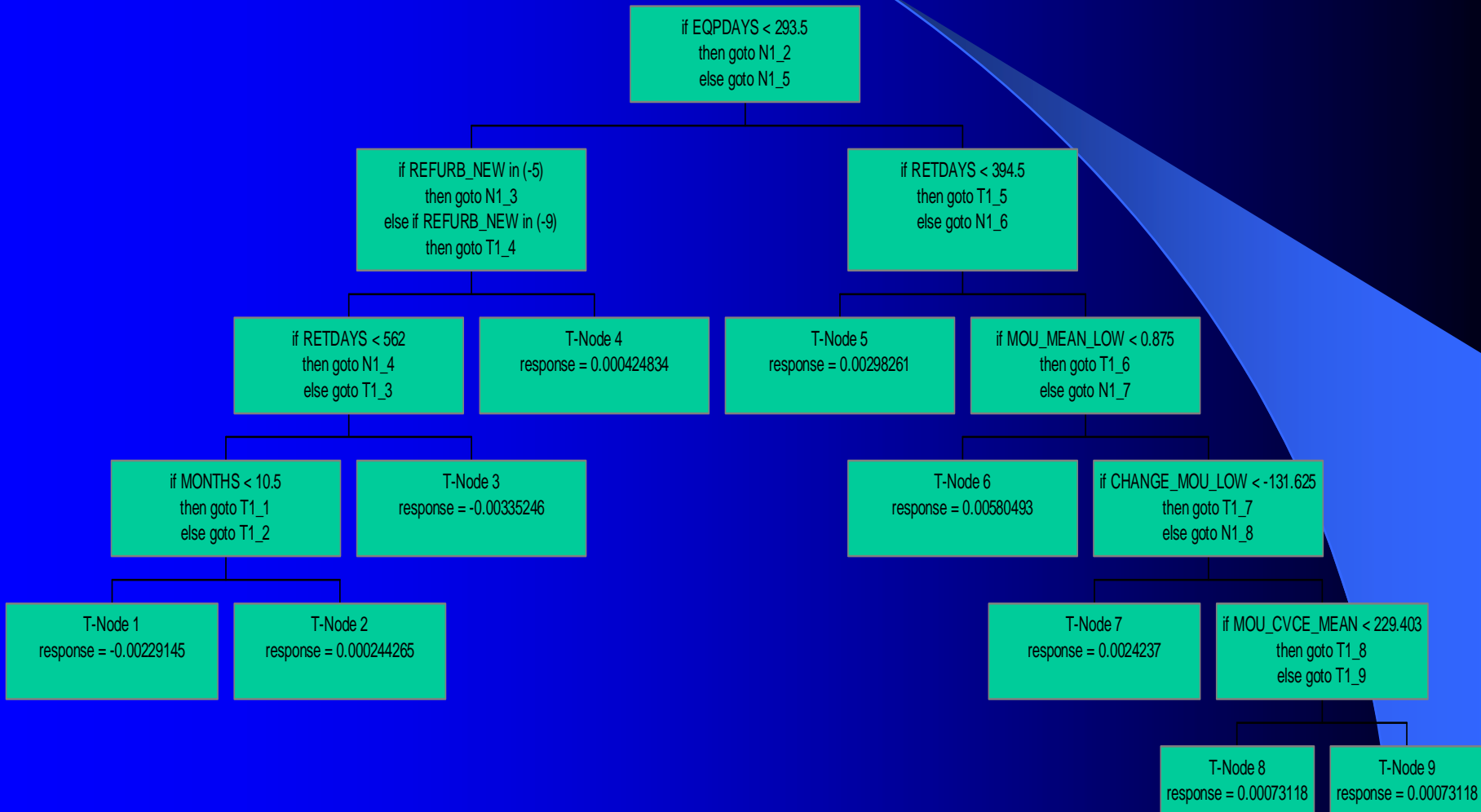
Treenet churn model first tree

Tree 1 of 2912



Treenet churn model second tree

Tree 2 of 2912



TreeNet Objective Function for Churn : Logistic Log-Likelihood LL

- $$LL = \sum_i \log \left(1 + e^{-2y_i F(x_i)} \right) = \sum_i l(y_i, F(x_i))$$

- The dependent variable, y , is coded (-1, +1)
- The target function, $F(x)$, is $1/2$ the log-odds ratio.
- F_0 is initialized to the log odds on the full training data set.
 - equivalent to fitting data to a constant.

$$F_o(X) = \frac{1}{2} \log \left(\frac{1+\bar{y}}{1-\bar{y}} \right)$$

Patient Learning Strategy: Key to TreeNet Success

- **Do not use all training data in any one iteration.**
 - Randomly sample from training data (we used a 50% sample).
- **Compute log-likelihood gradient for each observation.**

$$G(y_i, x_i) = \frac{\partial l(y_i, F_m(x_i))}{\partial F_m(x_i)} = \frac{2y_i}{1 + e^{2y_i F_m(x_i)}}$$

- **Build a K-node tree to predict $G(y_i, x_i)$.**
 - $K=9$ gave the best cross-validated results.
 - Important that trees be much smaller than the size of an optimal single CART tree.

Gradient Optimization

- Let

$$H(y_i, x_i) = - \frac{\partial^2 l(y_i, F_m(x_i))}{\partial F_m(x_i)^2}$$

- Update formula

$$H(y_i, x_i) = \frac{\partial \frac{2y_i}{1+e^{2y_i F_m(x_i)}}}{\partial F_m(x_i)} = \frac{4e^{2y_i F_m(x_i)}}{(1+e^{2y_i F_m(x_i)})^2} = |G(y_i, x_i)| (2 - |G(y_i, x_i)|)$$

- Repeat until T trees grown.
- Select the value of $m \leq T$ that produces the best fit to the test data.

$$F_{m+1}(x_i) = F_m(x_i) + \sum_n \beta_{mn} 1(x_i \in \Phi_{mn})$$

Newton-Raphson Step

- Compute γ_{mn} , a single Newton-Raphson step for β_{mn} .

$$\gamma_{mn} = \frac{\sum_{i \in \Phi_{mn}} G(y_i, x_i)}{\sum_{i \in \Phi_{mn}} H(y_i, x_i)}$$

- Use only a small fraction, ρ of γ_{mn} . ($\beta_{mn} = \rho\gamma_{mn}$).
- Apply the update formula

$$F_{m+1}(x_i) = F_m(x_i) + \sum_n \beta_{mn} 1(x_i \in \Phi_{mn})$$

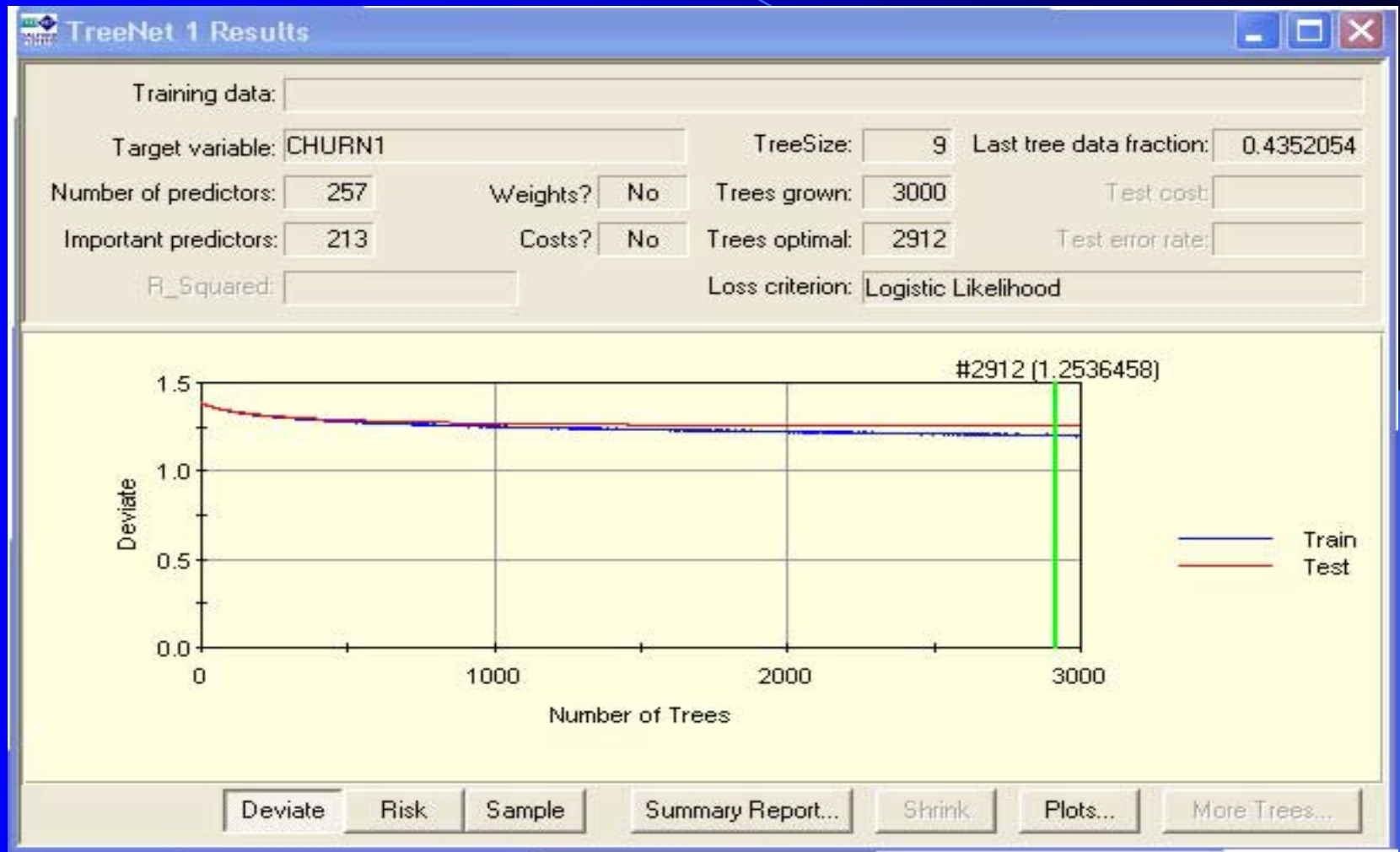
Learning Rates, Step Size, N Trees

- ρ is called the learning rate, T is the number of trees grown.
- The product ρT is the total learning.
 - Holding ρT constant, smaller ρ usually improves model fit to test data, but can require many trees.
- Reducing the learning rate tends to slowly increase the optimal amount of total learning.
- Very low learning rates can require many trees.
- Our CHURN models used values of ρ from .01 to .001.
- We used total learning of between 6 and 30.
- Our optimal models contained about 3,000 trees

The Salford CHURN Models

- All the models used to score the data for the entries used 9-node trees.
- Our final models used the following three combinations:
 - ($\rho = .001$; $T = 6000$; $\rho T = 6$);
 - ($\rho = .005$; $T = 2500$; $\rho T = 12.5$);
 - ($\rho = .01$; $T = 3000$; $\rho T = 30$)
- One entry was a single TreeNet model ($\rho = .01$; $T = 3000$; $\rho T = 30$).
 - In this range all models had almost identical results on test data.
 - The scores were highly correlated ($r \geq .97$).
 - Within this range, a higher ρT was the most important factor.
 - For models with $\rho T = 6$, the smaller the learning rate the better.

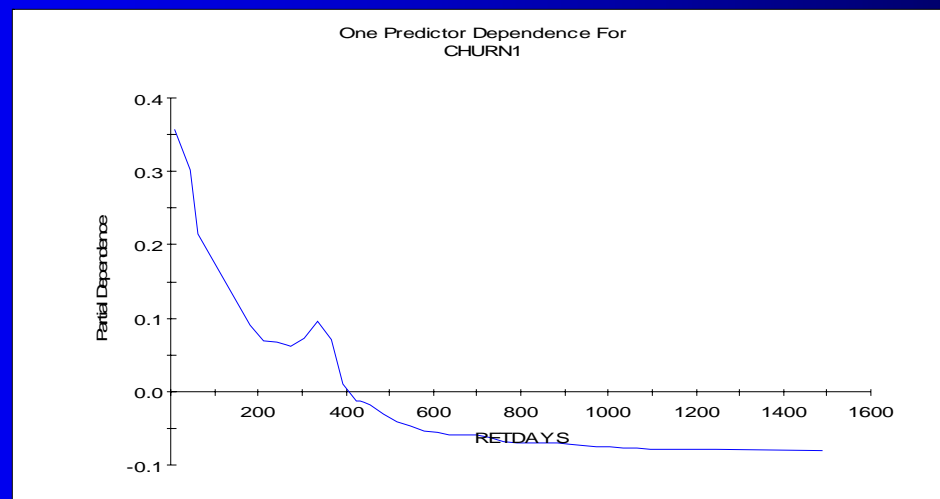
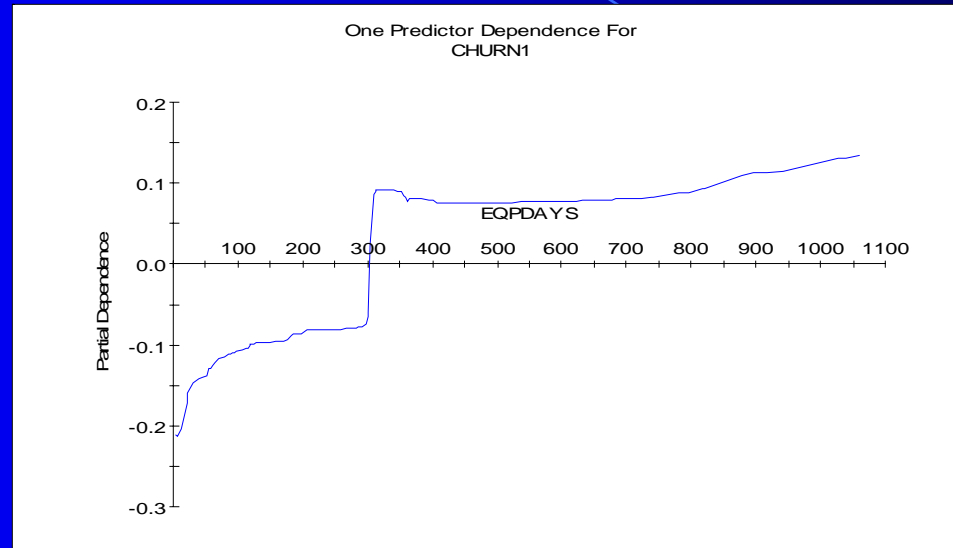
TreeNet Results Screen



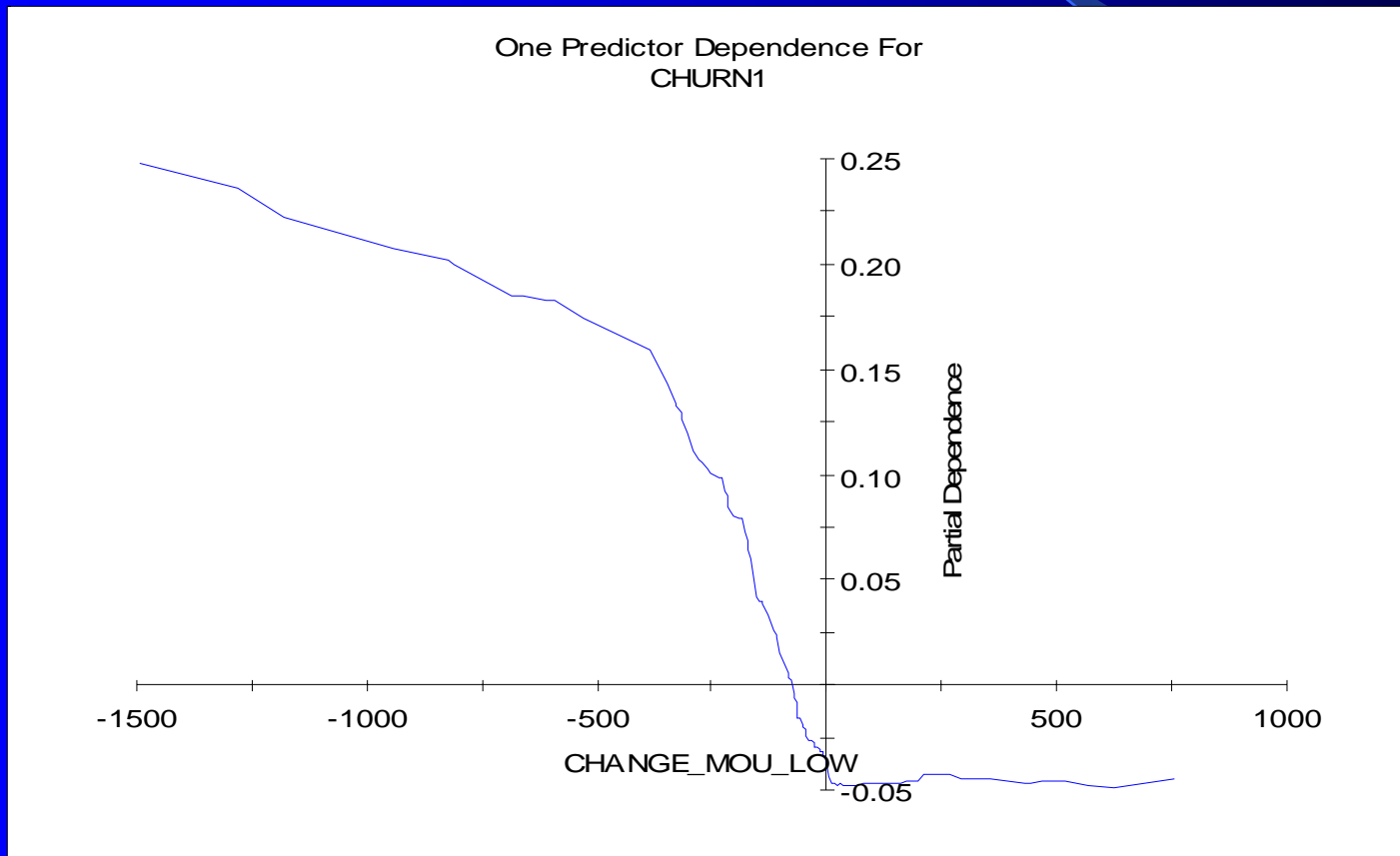
Model Results: Variable Importance

| Variable | Description | Score | |
|---------------|---|--------|-------|
| CRCLSCOD\$ | Credit Rating Grade (A-Z) | 100.00 | ***** |
| AREA\$ | Geographic Locale or Major City (19 levels) | 85.67 | ***** |
| ETHNIC\$ | Race/Origin (17 Levels) | 51.91 | ***** |
| EQPDAYS | Age of current handset | 46.39 | ***** |
| RETDAYS | Days since last retention call | 45.95 | ***** |
| CHANGE_MOU | Recent Change in Monthly Minutes | 37.51 | ***** |
| DWLLSIZE | Number of Households at address | 36.10 | ***** |
| MOU_MEAN | Lifetime average minutes usage | 35.40 | **** |
| OCCU1\$ | Occupation (Blue/White, Self) (22 levels) | 34.16 | **** |
| MONTHS | Length of service to date | 33.42 | **** |
| TOTMRC_RANGE | Range of monthly recurring charges | 31.38 | **** |
| CSANODE | CSA condensed to 8 levels (CART nodes) | 31.08 | **** |
| AVGQTY | Avg monthly calls (lifetime) | 26.35 | **** |
| MOU_CVCE_MEAN | Avg Monthly Minutes (completed voice) | 24.02 | *** |
| AVGMOU | Avg Monthly Minutes (lifetime) | 23.90 | *** |
| HND_PRICE | Hand Set Price | 23.43 | *** |

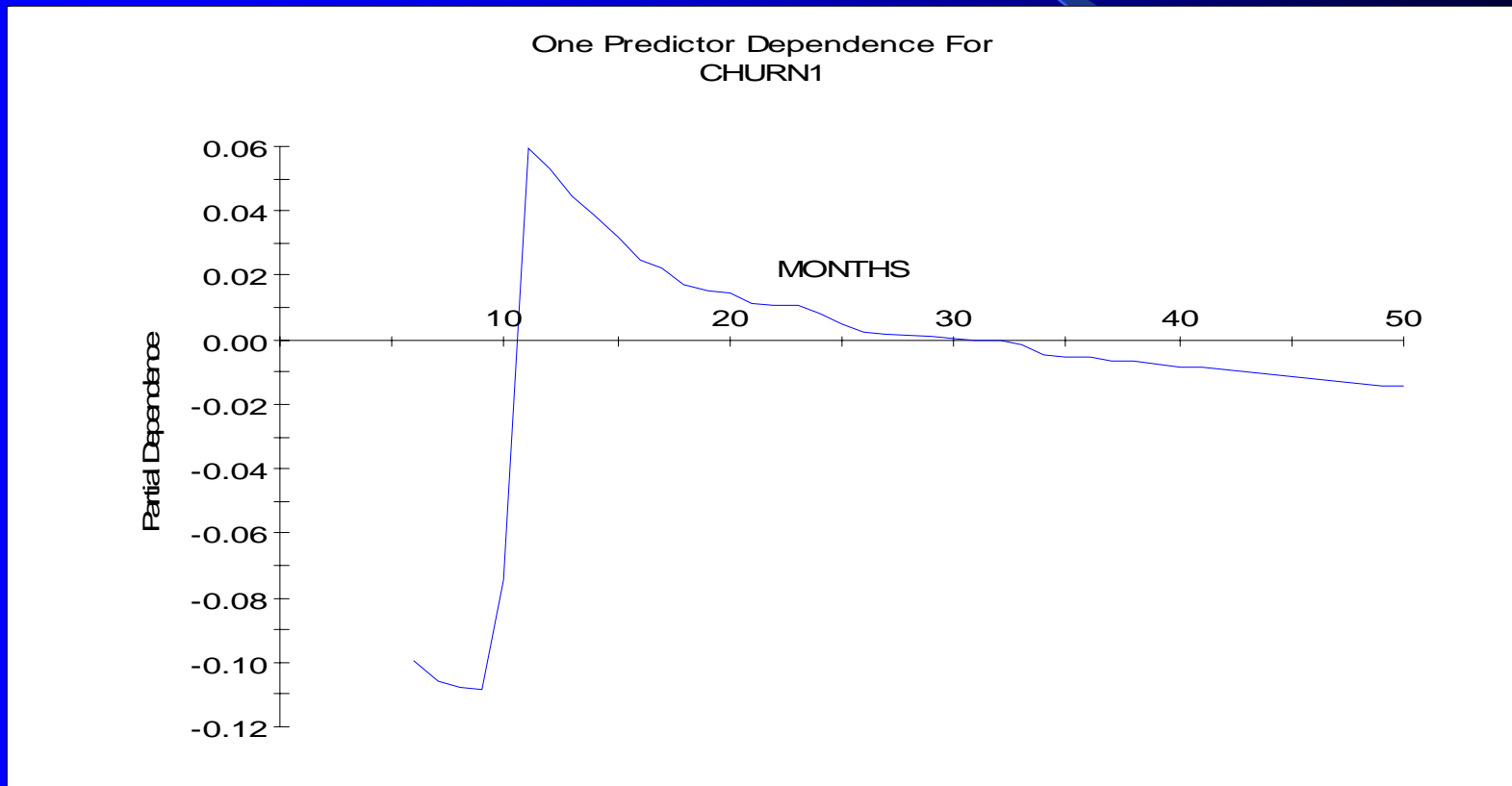
Impact of Hand Set Age and Time Since Retention Call



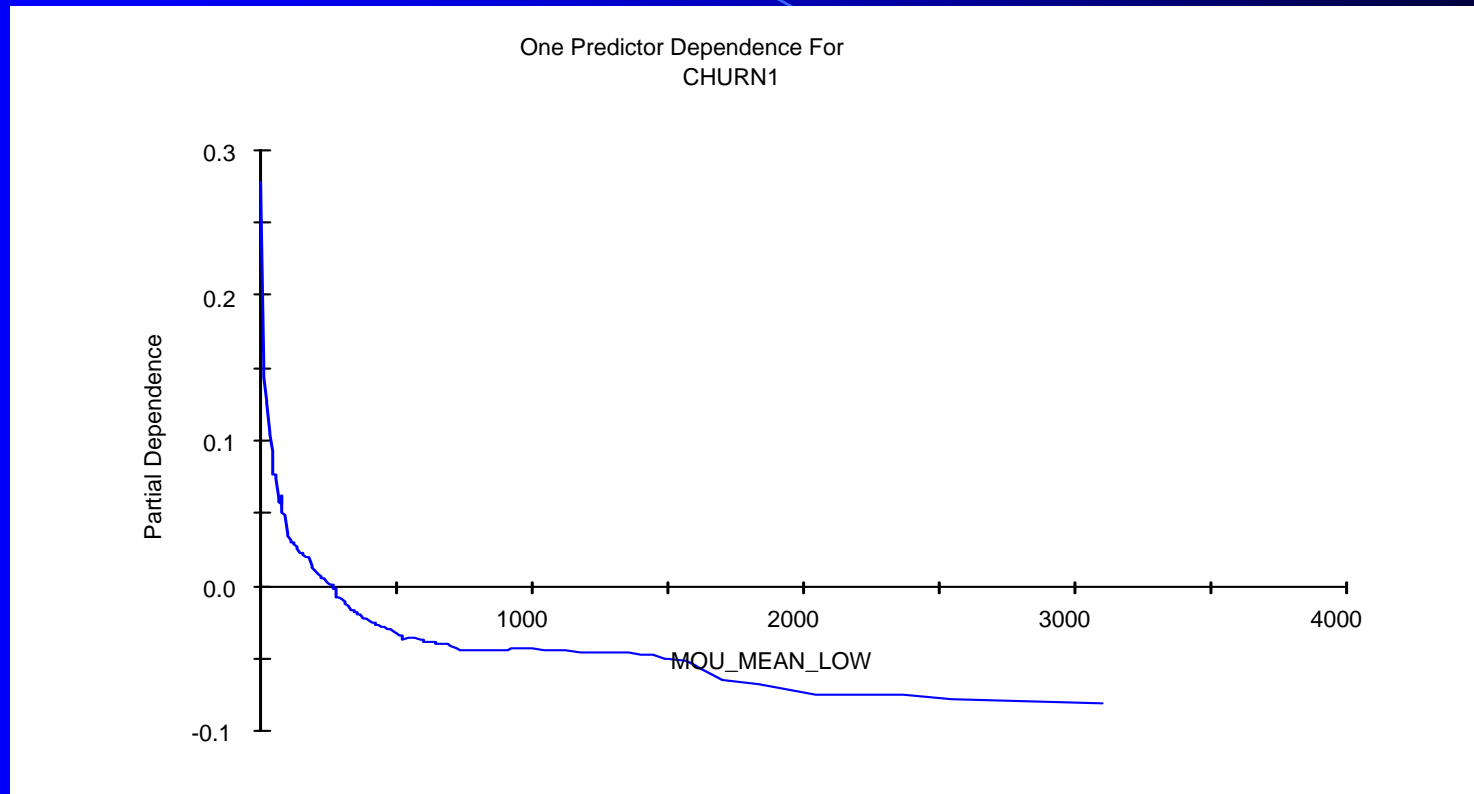
Effect of Recent Change in Minutes Usage



Effect of Tenure: The One-year Spike



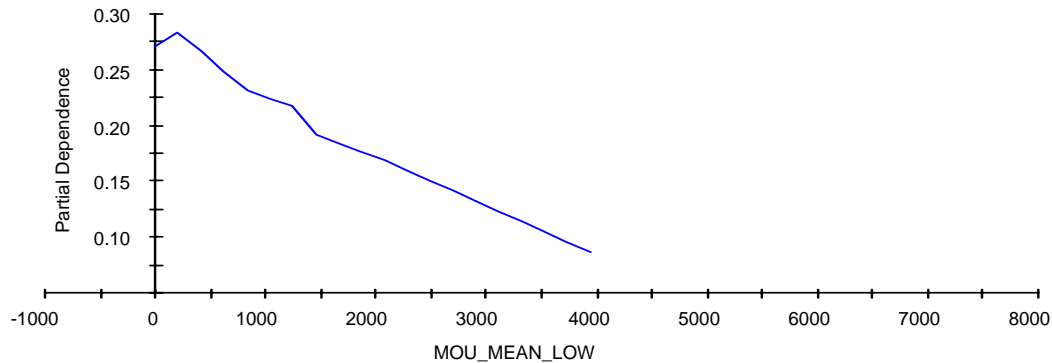
Prob of Churn vs Minutes



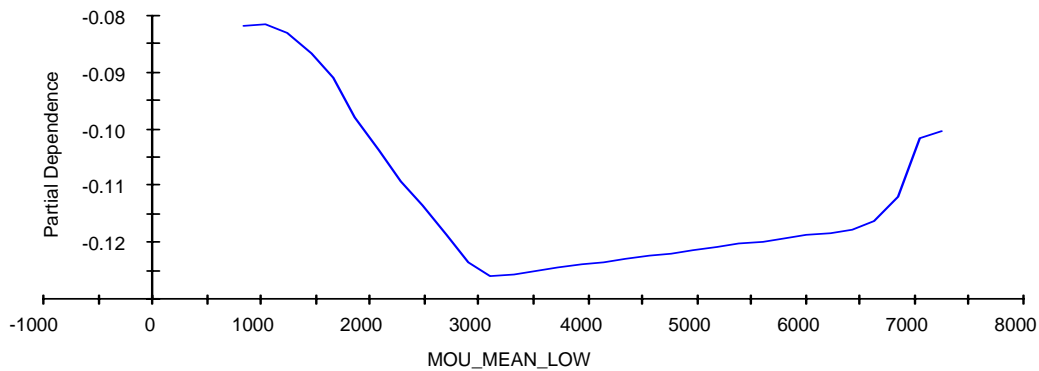
Unconditional; other variables varying in typical fashion

Interaction of Minutes and Change in Minutes

Two Variable Dependence for CHURN1; Slice CHANGE_MOU_LOW = -1675.891903119212
CHURN1



Two Variable Dependence for CHURN1; Slice CHANGE_MOU_LOW = 938.72975814036613
CHURN1



References

- **Friedman, J.H. (1999). Stochastic gradient boosting. Stanford: Statistics Department, Stanford University.**
- **Friedman, J.H. (1999). Greedy function approximation: a gradient boosting machine. Stanford: Statistics Department, Stanford University.**
- **Salford Systems (2002) TreeNet™ 1.0 Stochastic Gradient Boosting. San Diego, CA.**
- **Steinberg, D., Cardell, N.S., and Golovnya, M. (2003) Stochastic Gradient Boosting and Restrained Learning. Salford Systems discussion paper.**