Credit Limit Optimization (CLO) for Credit Cards

Vijay S. Desai
CSCC IX, Edinburgh
September 8, 2005
Agenda

- Background
  - Traditional approaches to credit limit management
  - History of different products/approaches
- Characteristics of the ideal solution
- Approaches to problem formulation
- Building good action-effect models
- Optimization problem
- Challenges in deploying the solution
- Optimization in other areas of credit card industry
- Q and A
Historical Credit Limit Change Programs

- Implemented in the form of decision trees/strategies
- Champion/Challenger framework for improving strategies over time
  - Randomly assign accounts to champion or challenger strategy
  - Measure performance over time
- Takes a six to twelve months to evaluate each challenger strategy
- A very small number of potential champion strategies can be tested at a given time
- Difficult to analyze why a particular challenger strategy worked
Efforts to Improve Credit Line Decisions

- By internal modeling groups of large banks
  - Optimization (e.g., JP Morgan Chase)
  - Markov decision processes (e.g., Bank One)
- By vendors
  - Greedy algorithms
  - Analytics heavy approaches
  - Optimization heavy approaches
  - Deterministic optimization
  - Robust optimization
Characteristics of Ideal CLO Solution

- Identify an optimal solution without lengthy champion/challenger iterations
- Achieve the full potential of each and every account relationship, not some segment level abstraction
- Optimize specific campaign goals such as profit or revenue or balance
- Accommodate operational as well as business constraints
- Factor in uncertainty in estimates and environment
- Examine multiple scenarios before committing to a final course of action
- Easy to deploy solutions
- Experimental design to explore new areas
Components of CLO Solution

- Data
- Action-Effect models
- Optimization
- Deployment
- Evaluation
Data Requirements for CLO

- Actions related data
  - Data from past campaigns
  - Results from systematically designed experiments

- Behavior related data
  - Statement data
  - Authorizations
  - Payments
  - Call-center data

- Organizational data
  - Business constraints data
  - Operations constraints data

- External data
  - Bureau data
  - Marketing data
Expanding Beyond the “Comfort Zone”

<table>
<thead>
<tr>
<th>Risk Score</th>
<th>Low</th>
<th>Medium</th>
<th>High</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Limit Utilization</td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td>Delinquency Status</td>
<td>Clean</td>
<td>Dirty</td>
<td>Clean</td>
</tr>
<tr>
<td>Champion Credit Line Inc.</td>
<td>0</td>
<td>500</td>
<td>0</td>
</tr>
<tr>
<td>Test Group 1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Test Group 2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Test Group 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Test Group 4</td>
<td>500</td>
<td>1000</td>
<td>500</td>
</tr>
<tr>
<td>Test Group 5</td>
<td>500</td>
<td>1500</td>
<td>1000</td>
</tr>
<tr>
<td>Test Group 6</td>
<td>500</td>
<td>2000</td>
<td>1500</td>
</tr>
</tbody>
</table>

- Use Design of Experiments concepts
  - Generate the most amount of information at lowest cost
- Necessary to explore regions never tested before
  - e.g., increases for accounts with risky scores, higher increases for low risk accounts
- Necessary to build better action-effect models
- Help respond to competitive pressures or more aggressive business goals
Randomized Block Design Details

- **Response variable:** Change in balance over a 6 month period.
  - Could be credit risk, attrition risk, revenue

- **Treatment levels** (500, 1000, 1500, 2000, 2500, 3000, 4000, 5000, 6000, 7000, 8000, 9000)
  - More levels allows one to explore a wider space

- **Blocks** (RL, RMCLDS, RMCLDD, RMCHDC, RMCHDD, RHCLDC, RHCLDD, RHCHDC, RHCHDD)
  - Blocking helps group similar accounts together reducing variation
  - Treatments within blocks randomly assigned
  - Blocking allows block versus treatment interaction

- Incomplete blocks, i.e., not all treatments repeated in all blocks due to business reasons

- Unbalanced design, i.e., each treatment does not appear the same number of times in each block
Action-Effect Models for CLO

- Use influence diagrams to visualize problem
- Predict effect of change in credit line on credit risk, attrition risk, revenue, and profit
- Segmentation choice crucial
  - Use the same segments as credit risk models
  - Use same segments as credit strategy
  - Using different segmentation in later stages could be problematic
- Accuracy of action-effect models determine the validity of optimization results
Data for Action-Effect Models

- Response is non-linear
- Low risk segments close to saturation point
- High risk segments show better response to increases
- Similar curves for credit risk and attrition risk
Enhancing Data for Action-Effect Models

- Traditional clustering: each data point assigned to a single class
  - Manual: segmentation, e.g. age
  - Data-driven: k-means
- Soft clustering: each data point belongs to all clusters in graded degree
  - Cluster membership determined by distance from center.
  - Data-driven: Cluster centers and shape updated intelligently
- Soft-clustering allows same account to be used in multiple action-effect models
Cluster Scoring

- **Action-effect models:**
  - Build models for change in credit risk, attrition risk, transaction volume and revolving balance, given a credit limit increase using account level information
  - Separate set of models for each segment or group of segments
  - Likely to be nonlinear models

- **Scoring:**
  - Score each individual account
  - Predict the effects (credit risk, attrition risk, profit/revenue, etc.) of the action/treatment (increase in credit limit)
  - Feed the results as coefficients into the optimization module

- **Other inputs for predicting profits/losses:**
  - Factors that go into calculating the predictions definable by the user including fixed cost per account for calculating profit, interchange fee, interest rate, late fees, annual fees, and over-limit fees
Optimization Problem Formulations

- **Non-linear programming formulations (A)**
  - Accounts for non-linear response to credit line increases
  - Even the simplest formulations are difficult to solve

- **Linear programming formulations (B, C)**
  - Only possible formulation at account level
  - Solutions might be difficult to interpret and deploy
  - Ignores uncertainty in coefficients

- **Robust optimization formulations (D)**
  - Takes uncertainty into account

- **Markov decision processes (E)**
  - Allow multiple credit line increases
Non-Linear Programming Example (A)

\[
\begin{align*}
\text{Max} & \quad \sum_i f(x_i) \\
\text{s.t.} & \quad \sum_i x_i \leq B \\
& \quad \sum_i g(x_i) \leq L \\
& \quad \sum_i f(x_i) \geq (1 + R) \sum_i x_i \\
& \quad x_i \geq 0
\end{align*}
\]

where \( x_i \) are the credit limit increases, \( f(x_i) \) is the profit for a given increase, \( g(x_i) \) is the loss for a given increase, \( B \) is the total credit increase budget, \( L \) is the total allowable losses, and \( R \) is the hurdle rate.

- Credit limit increases are a continuous variable
- Randomly choose a small number of accounts for optimization
- Use Lagrangian relaxation techniques
- Adding more constraints can make the solution more difficult
- Map optimal solution to a decision tree to score all accounts
- Deploying decision tree in lieu of solution can result in significant loss in benefit of the whole effort
Linear Programming  Example I (B)

Max $\sum_j \sum_i p_{ji} x_{ji}$

s.t. $\sum_j c_j \sum_i N_i x_{ji} \leq B$

$\sum_j \sum_i l_{ji} x_{ji} \leq L$

$\sum_j \sum_i p_{ji} x_{ji} \geq (1 + R) \left( \sum_j c_j \sum_i N_i x_{ji} \right)$

$\sum_j x_{ji} = 1 \forall i$

$x_{ji} \in \{0,1\}$

where $x_{ji}$ is 1 if account $i$ is given credit line $c_j$

$p_{ji}$ is the profit for a given increase, and

$l_{ji}$ is the loss a given increase

- Only discrete credit limit increases allowed
- Subset of LP problem has integer solutions most of the time
- Account level optimization possible
- Solve relaxed LP problem and check feasibility for remaining constraints
- No need to map optimal solution to a score
Linear Programming Example II (C)

\[
\begin{align*}
\text{Max} & \quad \sum_{i} \sum_{j} \sum_{x_{ji}} p_{x_{ji}} x_{ji} \\
\text{s.t.} & \quad \sum_{j} c_{j} \sum_{i} N_{i} x_{ji} \leq B \\
& \quad \sum_{i} \sum_{j} \sum_{l_{x_{ji}}} x_{ji} \leq L \\
& \quad \sum_{i} \sum_{j} \sum_{p_{x_{ji}}} x_{ji} \geq (1 + R) \left( \sum_{j} c_{j} \sum_{i} N_{i} x_{ji} \right) \\
& \quad \sum_{j} x_{ji} = 1 \text{ for each } i \\
& \quad 1 \geq x_{ji} \geq 0 \\
\end{align*}
\]

where \( x_{ji} \) are the fraction of accounts in segment \( i \) with credit limit increase \( c_{j} \),

\( p_{x_{ji}} \) is the profit in time period \( t \) for a given increase, and

\( l_{x_{ji}} \) is the loss in time period \( t \) for a given increase

- Only discrete credit limit increases allowed
- Segment level optimization
- Random fraction of accounts in a segment get a particular increase
- No need to map optimal solution to a score
- Predicting profits and losses over multiple time periods difficult
Robust Optimization Example (D)

\[ \text{Max} \quad \sum_k \left( \sum_i \sum_j \sum_l P_{kij} x_{ji} \right) / M \]

\[ s.t. \quad \sum_j c_j \sum_i N_i x_{ji} \leq B \]
\[ \sum_i \sum_j \sum_l l_{kij} x_{ji} \leq L \quad \forall \ k \]
\[ \sum_i \sum_j \sum_l P_{kij} x_{ji} \geq (1 + R) \left( \sum_j c_j \sum_i N_i x_{ji} \right) \quad \forall \ k \]
\[ \sum_j x_{ji} = 1 \quad \text{for each} \ i \]
\[ 1 \geq x_{ji} \geq 0 \]

where \( x_{ji} \) are the fraction of accounts in segment \( i \) with credit limit increase \( c_j \),
\( P_{kij} \) is the profit for simulation \( k \) for a given increase, and
\( l_{kij} \) is the loss for simulation \( k \) for a given increase.

- Perform \( M \) simulations to reflect uncertainty in profit and loss coefficients
- Objective function is an expected value
- Loss and hurdle rate constraints might not be satisfied for all scenarios
- Realized profit and loss values more likely to match optimization solution values
Markov Decision Processes Example (E)

\[ V_t(c,i) = \left\{ \begin{array}{ll} \max_{c_j} \left\{ r(c + c_j,i) + \beta \sum_{j} p(c + c_j,i;l)V_{t+1}(c + c_j,l) \right\} & \text{if } t = \text{update epoch} \\ r(c,i) + \beta \sum_{j} p(c,i;l)V_{t+1}(c,l) & \text{otherwise} \end{array} \right. \]

\[ V_T(c,i) = r(c,i) \]

where the account has credit line \( c \), is in one of the segments \( i \), single period profit is \( r(c,i) \), the transition matrix is \( p(c,i;l) \), the value function is \( V_t(c,i) \), the time horizon is \( T \), and the one-period discount factor is \( \beta \)

- Multiple credit line increases over time horizon \( T \)
- Takes uncertainty into account
- Can still include constraints, e.g., hurdle rate
- Curse of dimensionality, e.g., 20 treatments and 9 segments gives 180 states, 32,400 cell transition matrix
- Markov assumptions might be violated, e.g., new accounts, balance transfers, other treatments
Optimization Formulations Critique

- **(A)** maintains the non-linear nature of response to credit line changes, and continuous change in credit limits.
  - Useful for testing to what extent discretizing the search space in (B-E) results in suboptimal solutions

- **(B)** is the only formulation with possible account level optimization, albeit after ignoring some constraints.
  - Useful for testing to what extent sampling (A) and segment level search (C-E) results in suboptimal solutions

- **(D)** only formulation that take into account the uncertainty in profit as well as loss in response to decisions
  - Useful for testing the impact of uncertainty

- **(E)** only formulation that takes into account multi-period decision making. (C-D) look at multi-period profits and losses, but do not allow multiple credit line changes
  - Useful for testing the value of multiple credit line changes
Optimization Formulations Experience

- Accurate estimation of coefficients crucial
  - Inaccuracies can completely negate the optimization approach
  - Design of experiments to collect additional data extremely useful
  - Using all data sources, including transactional data, extremely useful in building more accurate models

- Accounting for uncertainty crucial
  - Realized profits and losses in production much closer to those predicted by robust optimization as compared to deterministic optimization
  - Also important to take into account correlations between all sources of uncertainty

- Segmentation scheme crucial
  - Impacts accuracy of action-effect models
  - Impacts search space explored

- SAS solution does all of the above
Profit Frontier

![Graphical representation of Profit Frontier]

Copyright © 2003, SAS Institute Inc. All rights reserved.
Impact of Uncertainty on Objective Function

- D.O. generates higher expected total profit.
- D.O. has much wider spread when evaluated w/ various scenarios.
- S.O. generated smaller total profit, but higher profit per account.
- S.O. generates a tighter distribution overall.
- Results from S.O. gets better as the variance in the parameters increases.
Impact of Uncertainty on Constraints

- S.O. is more robust than D.O.
- When simulated with various scenarios based on the covariance structure in the coefficients,
  - D.O. violates the constraints in 51% of the out-of-sample evaluations.
  - S.O. violates the constraints 5.5% of the simulation.
Deployment Options for CLO

- Optimization solution output deployed as a list of account numbers with recommended credit limit changes
  - Does not dilute the optimization results
  - Difficult to use in many situations, e.g., production system constraints, accounts not included in the optimization exercise

- Optimization solution used to create a decision tree. Decision tree deployed to score all accounts and determine credit limit changes in production
  - Mapping solution to decision tree can significantly dilute the results
  - Only option when optimization done for a small sample of accounts, or if accounts were not included in the optimization exercise
  - Easier to compare with traditional approaches, or fit into a champion/challenger methodology
Evaluation for CLO

- Important to close the loop
  - Compare production and modeling results
  - Compare results with older strategies

- Use results for planning
  - Use results to collect new data and fill more gaps
  - Interactions with other treatments, loss forecasting
Q & A

- Thank you for the opportunity to present