Comparing regret minimization and utility maximization for route choice using the recursive logit model

ICMC, The University of Texas at Austin, U.S
May 10-13, 2015

Tien Mai, Emma Frejinger and Fabian Bastin

Department of Computer Science and Operational Research
Université de Montréal and CIRRELT, Canada
# Table of Contents

1. Introduction

2. RRM and Recursive Logit Model

3. Specifications of regret functions

4. Maximum likelihood estimation

5. Numerical results

6. Conclusion
Random utility maximization and random regret minimization frameworks

- Random utility maximization (RUM)
  \[ i^* = \arg\max_{i \in C_n} \{ V_{ni} + \epsilon_{ni} \} \]
  \[ V_{ni} = \beta x_{ni} \]

- Random regret minimization (RRM) (Chorus, 2010)
  \[ i^* = \arg\min_{i \in C_n} \{ R_{ni} + \epsilon_{ni} \} \]
  \[ R_{in} = \sum_{j \neq i} \sum_{t} \ln \left( 1 + e^{\beta_t (x_{jn(t)} - x_{in(t)})} \right) \]
  \[ t \text{ is an attribute} \]
Prato (2012) analyses the estimation results of path-based models using the RRM framework proposed by Chorus (2010). He finds that in an experimental setting, the parameter estimates have the wrong signs when irrelevant alternatives are included in the choice sets.

RRM model can be expensive to estimate when choice sets become very large and every alternative is compared with every other alternative in the choice set in terms of every attribute.

Fosgerau et al. (2013) proposed the link-based recursive logit (RL) model which is consistently estimated without sampling any choice sets of paths and straightforward for prediction.

RL model is based on infinite choice sets, but the choice set at choice stage is small (number of outgoing links in road networks).
Highlights

▶ We adapt and propose two specifications for the regret function based on the RRM model proposed by Chorus (2014)
▶ Estimation methods for the regret-based models with the RL models
▶ Estimation and cross-validation results for a real network with over 3000 nodes and 7000 links
▶ We compare the results with the RUM-based models
Recursive logit with regret-based models

- Path choices are modeled as sequences of link choices using a dynamic discrete choice framework.
- At each choice stage decision makers maximize the sum of the random utility of outgoing links (instantaneous utility) and the expected maximum utility from the sink node of the links to the destination (value function).
- We can formulate the regret models in a similar way.
- Decision makers minimize the sum of the random regret of outgoing links and the expected minimum regret from the sink node of the links to the destination.
Recursive logit with regret-based models

\[ r(a|k)/v(a|k) \]

\[ R(a)/V(a) \]

\[ A(k) \]

\[ d \]

\[ k \]
Generalized random regret minimization model (Chorus, 2014)

- Generalized Random Regret Minimization (GRRM) model (Chorus, 2014)

\[
   r(a|k) = \sum_{a' \in A(k), a' \neq a} \sum_t \ln \left( \lambda_t + e^{\beta_t (x(a'|k)_t - x(a|k)_t)} \right)
\]

- \( x(a|k) \) is a vector of attribute, \( t \) is attribute
- \( A(k) \) may contain only one link. GRRM (or other existing RRM models) would assign a regret zero, which would cause numerical issues for the RL model.
Extended random regret minimization model

- Extended random regret minimization (ERRM) model

\[ r(a|k) = \sum_{a' \in A(k)} \sum_t \ln \left( \lambda_t + e^{\beta_t \left( x(a'|k)_t - x(a|k)_t \right)} + \delta_t x(a'|k)_t \right) \]

- If \( \delta_t > 0 \), the impact of the non-chosen alternatives becomes larger and if \( \delta_t < 0 \), it is smaller
Averaged random regret minimization model

- Model generalizes the RUM-based RL model in Fosgerau et al., 2013
- A normalization factor is used, the regret is averaged over all the alternatives
- Averaged Random Regret Minimization (ARRM) model

\[
 r(a|k) = \frac{1}{|A(k)|} \sum_{a' \in A(k)} \sum_t \ln \left( \lambda_t + e^{\beta_t \left( x(a'|k)_t - x(a|k)_t \right)} + \delta_t x(a'|k)_t \right)
\]

- Specifying \( \lambda_t = 0, \delta_t = -\beta_t \) \( \forall t \), \( r(a|k) = -\beta x(a|k) \), the ARRM model is identical to the RUM-based RL model
Maximum likelihood estimation

- Nested fixed point algorithm (Rust, 1987): combines an outer iterative non-linear optimization algorithm for searching over the parameter space, with an inner algorithm for solving the expected minimum regrets
- The expected minimum regrets can be computed by solving linear systems
- The RRM models require constraints $0 \leq \lambda_t \leq 1$, $\forall t$. We use the interior point algorithm with BFGS to solve the constrained problem
Network

The Borlänge network (used in Fosgerau et al., 2013)

- 3077 nodes, 7459 links, 21452 link pairs
- Travel times are assumed static and deterministic
- 1832 trips corresponding to simple paths with a minimum of 5 links
- 466 destinations, 1420 different origin-destination (OD) pairs and more than 37,000 link choices
- Attributes: travel time (TT), left turn (LT), link constant (LC), u-turn (UT)
Parameter estimates

<table>
<thead>
<tr>
<th>Parameters</th>
<th>GRRM</th>
<th>ARRM</th>
<th>ERRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{\beta}_{TT}$</td>
<td>-0.15</td>
<td>-1.92</td>
<td>-0.37</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>-11.46</td>
<td>-8.98</td>
<td>-4.05</td>
</tr>
<tr>
<td>$\hat{\beta}_{LT}$</td>
<td>-0.34</td>
<td>-1.80</td>
<td>-0.31</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>-15.36</td>
<td>-4.43</td>
<td>-3.84</td>
</tr>
<tr>
<td>$\hat{\beta}_{UT}$</td>
<td>-5.89</td>
<td>-7.32</td>
<td>-5.32</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>-10.32</td>
<td>-0.11</td>
<td>-2.85</td>
</tr>
<tr>
<td>$\hat{\beta}_{LC}$</td>
<td>12.92</td>
<td>99.99</td>
<td>23.18</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>7.77</td>
<td>2.77</td>
<td>6.11</td>
</tr>
</tbody>
</table>
## Parameter estimates

<table>
<thead>
<tr>
<th>Parameters</th>
<th>ARRM</th>
<th>ERRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{\delta}_{TT}$</td>
<td>3.75</td>
<td>1.22</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>8.18</td>
<td>5.69</td>
</tr>
<tr>
<td>$\hat{\delta}_{LT}$</td>
<td>0.12</td>
<td>0.09</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>0.17</td>
<td>0.89</td>
</tr>
<tr>
<td>$\hat{\delta}_{UT}$</td>
<td>7.16</td>
<td>4.75</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>0.13</td>
<td>3.62</td>
</tr>
<tr>
<td>$\hat{\delta}_{LC}$</td>
<td>-7.16</td>
<td>-1.44</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>-0.11</td>
<td>-0.84</td>
</tr>
</tbody>
</table>
### Parameter estimates

<table>
<thead>
<tr>
<th>Parameters</th>
<th>GRRM</th>
<th>ARRM</th>
<th>ERRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{\lambda}_{TT}$</td>
<td>8.13e-6</td>
<td>0.37</td>
<td>1.00</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>1.80e-4</td>
<td>1.20</td>
<td>2.55</td>
</tr>
<tr>
<td>$\hat{\lambda}_{LT}$</td>
<td>7.26e-6</td>
<td>1.00</td>
<td>8.29e-5</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>1.17e-4</td>
<td>2.24</td>
<td>5.53e-5</td>
</tr>
<tr>
<td>$\hat{\lambda}_{UT}$</td>
<td>0.76</td>
<td>0.01</td>
<td>1.04e-4</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>17.86</td>
<td>0.00</td>
<td>1.44e-4</td>
</tr>
<tr>
<td>$\hat{\lambda}_{LC}$</td>
<td>0.46</td>
<td>0.58</td>
<td>0.48</td>
</tr>
<tr>
<td>Rob. t-test(0)</td>
<td>21.62</td>
<td>0.02</td>
<td>0.59</td>
</tr>
</tbody>
</table>
## Final log-likelihoods comparison

<table>
<thead>
<tr>
<th></th>
<th>RL</th>
<th>RL-LS</th>
<th>NRL</th>
<th>NRL-LS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong># parameters</strong></td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td><strong>Final log-likelihood</strong></td>
<td>-6303.9</td>
<td>-6045.6</td>
<td>-6187.9</td>
<td>-5952.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>GRRM</th>
<th>ARRM</th>
<th>ERRM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong># parameters</strong></td>
<td>8</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td><strong>Final log-likelihood</strong></td>
<td>-7931.6</td>
<td>-5661.6</td>
<td>-5500.4</td>
</tr>
</tbody>
</table>
Prediction comparison

- Cross validation approach: 80% for estimation, 20% for prediction
- 40 holdout samples of the same size taken from the real sample
- Using loss functions

\[
\overline{err}_p = \frac{1}{p} \sum_{i=1}^{p} \left( - \sum_{\sigma \in HS_i} \ln P(\sigma, \hat{\beta}_i) \right) \quad \forall 1 \leq p \leq 40
\]

\( HS_i \) is the holdout sample \( i \) and \( \hat{\beta}_i \) is the parameter estimates for estimation sample \( i \)
Prediction comparison
Conclusions

- Based on this data set, the ERRM model performs the best (it also has a higher final log-likelihood value in the estimation) and the performance of the GRRM model is worse.
- RRM rule may be an interesting avenue for route choice modeling, however
  - Estimation and application of the RRM based models may computationally time consuming
  - The interpretation of the parameter estimates are less straightforward than RUM-based models
Thanks for your attention!