Nonparametric computation of survival functions in the presence of interval censoring

Stephen M. Taylor

MSc Candidate
Supervisor: Dr Yong Wang
Department of Statistics
The University of Auckland
stay020@aucklanduni.ac.nz

1 September 2008
Research Aims

- Create a robust algorithm for solving the NPMLE problem
- One that is fastest in all circumstances

Adaptive Constrained Newton Method (ACNM)
The HALT Study
ACNM Algorithm
Summing Up

Mean times, for n=400

Proportion of Exact Observations
Computation Time (s)
CNM
EMICM(DR)
SBN(DR)

Proportion of Exact Observations
Computation Time (s)
CNM
EMICM(DR)
SBN(DR)

and n=1600
Mean times, for $n=400$

![Graph showing computation times for $n=400$.](image1)

and $n=1600$

![Graph showing computation times for $n=1600$.](image2)
Survival Analysis

- Time to event data
Survival Analysis

- Time to event data
- Want to model the distribution of times to 'failure'
Survival Analysis

- Time to event data
- Want to model the distribution of times to ‘failure’
- Interested in the survival function, $S(t) = P(T > t)$
Survival Analysis

- Time to event data
- Want to model the distribution of times to ‘failure’
- Interested in the survival function, $S(t) = P(T > t)$
- Example: Time to healing
Censoring

- Time of event may not be directly measurable
Censoring

- Time of event may not be directly measurable
- Check periodically to see if it has occurred
Censoring

- Time of event may not be directly measurable
- Check periodically to see if it has occurred
- Example: healing occurred some time between doctor visits
Censoring

- Time of event may not be directly measurable
- Check periodically to see if it has occurred
- Example: healing occurred some time between doctor visits
- The event may never occur for some subjects
Censoring

- Time of event may not be directly measurable
- Check periodically to see if it has occurred
- Example: healing occurred some time between doctor visits
- The event may never occur for some subjects
- Example: end of study or “lost to followup”
Interval Censoring

- Event times are not known exactly, only within intervals
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
- Interval censored: event occurred somewhere in $(t_L, t_R]$
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
- Interval censored: event occurred somewhere in \((t_L, t_R]\)
- Right censored: \((t_L, \infty)\)
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
- Interval censored: event occurred somewhere in $(t_L, t_R]$
- Right censored: $(t_L, \infty)$
- Left censored: $(0, t_R]$
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
- Interval censored: event occurred somewhere in $(t_L, t_R]$.
- Right censored: $(t_L, \infty)$
- Left censored: $(0, t_R]$.
- Exact observation: event occurred at time $t$. 
Interval Censoring

- Event times are not known exactly, only within intervals
- Perhaps no event time is observed exactly
- Interval censored: event occurred somewhere in \((t_L, t_R]\)
- Right censored: \((t_L, \infty)\)
- Left censored: \((0, t_R]\)
- Exact observation: event occurred at time \(t\)
- Call these intervals \(O_i\) for \(i = 1, \ldots, n\)
Why Nonparametric?

- Let the data speak for itself
Why Nonparametric?

- Let the data speak for itself
- Don’t make assumptions about the distribution
Why Nonparametric?

- Let the data speak for itself
- Don’t make assumptions about the distribution
- Maximise the likelihood
Why Nonparametric?

- Let the data speak for itself
- Don’t make assumptions about the distribution
- Maximise the likelihood
- Explore the data before choosing a parametric model
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
- Potential support intervals
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
- Potential support intervals
- Only use maximal cliques
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
- Potential support intervals
- Only use maximal cliques
- Support set: $I_j$ for $j = 1, \ldots, m$
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
- Potential support intervals
- Only use maximal cliques
- Support set: $I_j$ for $j = 1, \ldots, m$
- The clique matrix $A_{n \times m}$ gives $\delta_{ij}$ membership of each $O_i$ in each $I_j$
The NPMLE Survival Function with Interval Censored Data

- Partition the positive real line
- All unique values of $t_L$ and $t_R$
- Potential support intervals
- Only use maximal cliques
- Support set: $I_j$ for $j = 1, \ldots, m$
- The clique matrix $A_{n \times m}$ gives $\delta_{ij}$ membership of each $O_i$ in each $I_j$
- NPMLE assigns probability mass to each support interval
Likelihood Function for the NPMLE

- Likelihood of an interval $(t_1, t_2]$ is $S(t_1) - S(t_2)$
Likelihood Function for the NPMLE

- Likelihood of an interval \((t_1, t_2]\) is \(S(t_1) - S(t_2)\)
- Assign probability \(p_j\) to support interval \(I_j\)
Likelihood Function for the NPMLE

- Likelihood of an interval \((t_1, t_2]\) is \(S(t_1) - S(t_2)\)
- Assign probability \(p_j\) to support interval \(I_j\)
- Probability of observation \(O_i\) using \(A\) and \(p\)
Likelihood Function for the NPMLE

- Likelihood of an interval \((t_1, t_2]\) is \(S(t_1) - S(t_2)\)
- Assign probability \(p_j\) to support interval \(I_j\)
- Probability of observation \(O_i\) using \(A\) and \(p\)
- Take logs and add them up
Likelihood Function for the NPMLE

- Likelihood of an interval \((t_1, t_2]\) is \(S(t_1) - S(t_2)\)
- Assign probability \(p_j\) to support interval \(l_j\)
- Probability of observation \(O_i\) using \(A\) and \(p\)
- Take logs and add them up
- Goal: find \(\hat{p} \in \mathbb{R}^m\) to maximise \(\ell(\hat{p})\)
Likelihood Function for the NPMLE

- Likelihood of an interval \((t_1, t_2]\) is \(S(t_1) - S(t_2)\)
- Assign probability \(p_j\) to support interval \(I_j\)
- Probability of observation \(O_i\) using \(A\) and \(p\)
- Take logs and add them up
- Goal: find \( \hat{p} \in \mathbb{R}^m \) to maximise \( \ell(\hat{p}) \)
- Subject to: \( \hat{p} \geq 0 \) and \( \hat{p}^T \mathbf{1} = 1 \)
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
- Effect of Manuka Honey dressings for treatment of leg ulcers
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
- Effect of Manuka Honey dressings for treatment of leg ulcers
- Participants assessed weekly and also at a 12-week follow-up
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
- Effect of Manuka Honey dressings for treatment of leg ulcers
- Participants assessed weekly and also at a 12-week follow-up
- Nurse changes dressing and assesses healing status
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
- Effect of Manuka Honey dressings for treatment of leg ulcers
- Participants assessed weekly and also at a 12-week follow-up
- Nurse changes dressing and assesses healing status
- Event times cannot be observed exactly
Honey as Adjuvant Leg Ulcer Therapy (HALT)

- Randomised Clinical Trial, 368 participants
- Clinical Trials Research Unit in Auckland
- Effect of Manuka Honey dressings for treatment of leg ulcers
- Participants assessed weekly and also at a 12-week follow-up
- Nurse changes dressing and assesses healing status
- Event times cannot be observed exactly
- Thanks to Andrew Jull and Varsha Parag of CTRU for providing the data
Censor Intervals for each Participant

Time (weeks) to Healing

Subject
The HALT Study

ACNM Algorithm

Summing Up
Existing Algorithms for finding the NPMLE

- The Icens package in R provides five algorithms:
Existing Algorithms for finding the NPMLE

- The Icens package in R provides five algorithms:
  - EM, ISDM, EMICM, VEM and PGM
Existing Algorithms for finding the NPMLE

- The Icens package in R provides five algorithms:
  - EM, ISDM, EMICM, VEM and PGM
  - Subspace-based Newton method (Dümbgen et al. 2006)
Existing Algorithms for finding the NPMLE

- The `Icens` package in R provides five algorithms:
  - EM, ISDM, EMICM, VEM and PGM
- Subspace-based Newton method (Dümbgen et al. 2006)
- Wang (2008) introduced:
Existing Algorithms for finding the NPMLE

- The Icens package in R provides five algorithms:
  - EM, ISDM, EMICM, VEM and PGM
- Subspace-based Newton method (Dümbgen et al. 2006)
- Wang (2008) introduced:
  - Constrained Newton Method
Existing Algorithms for finding the NPMLE

- The Icens package in R provides five algorithms:
  - EM, ISDM, EMICM, VEM and PGM
- Subspace-based Newton method (Dümbgen et al. 2006)
- Wang (2008) introduced:
  - Constrained Newton Method
  - Dimension-reduced approach to improve any algorithm
Times to compute the NPMLE survival function for 100 Bootstrap samples of the HALT data using:

- EMICM, PGM and VEM from the Icens package
- Methods SBN(DR) and EMICM(DR) from Wang (2008)
- The new ACNM algorithm (and CNM)

<table>
<thead>
<tr>
<th>Method</th>
<th>Time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMICM</td>
<td>113.03</td>
</tr>
<tr>
<td>PGM</td>
<td>791.00</td>
</tr>
<tr>
<td>VEM</td>
<td>610.42</td>
</tr>
<tr>
<td>SBN(DR)</td>
<td>14.34</td>
</tr>
<tr>
<td>EMICM(DR)</td>
<td>26.93</td>
</tr>
<tr>
<td>ACNM</td>
<td>9.41</td>
</tr>
</tbody>
</table>
Problems with Existing Algorithms

- Some are very slow and may fail to converge
Problems with Existing Algorithms

- Some are very slow and may fail to converge
- No algorithm outperforms the others in all situations
Problems with Existing Algorithms

- Some are very slow and may fail to converge
- No algorithm outperforms the others in all situations
- Inefficient use of Hessian matrix or gradient
Problems with Existing Algorithms

- Some are very slow and may fail to converge
- No algorithm outperforms the others in all situations
- Inefficient use of Hessian matrix or gradient
- Best choice depends on size of dataset and proportion of exact observations
Constrained Newton Method

- Calculates gradient $S$ of $\ell(p)$ at current estimate $p$
Constrained Newton Method

- Calculates gradient $S$ of $\ell(p)$ at current estimate $p$
- Makes use of mixture structure of solution
Constrained Newton Method

- Calculates gradient \( S \) of \( \ell(p) \) at current estimate \( p \)
- Makes use of mixture structure of solution
- Uses NNLS to find new estimate of \( p \)
Constrained Newton Method

- Calculates gradient $S$ of $\ell(p)$ at current estimate $p$
- Makes use of mixture structure of solution
- Uses NNLS to find new estimate of $p$
- Computation time of NNLS is of order $O(nm^2)$
Constrained Newton Method

- Calculates gradient $S$ of $\ell(p)$ at current estimate $p$
- Makes use of mixture structure of solution
- Uses NNLS to find new estimate of $p$
- Computation time of NNLS is of order $O(nm^2)$
- Very fast for fully censored datasets
Constrained Newton Method

- Calculates gradient $S$ of $\ell(p)$ at current estimate $p$
- Makes use of mixture structure of solution
- Uses NNLS to find new estimate of $p$
- Computation time of NNLS is of order $O(nm^2)$
- Very fast for fully censored datasets
- Can be slow in cases with many exact observations
Adaptive CNM

- Uses a divide and conquer approach
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
- Adapts to the data to make efficient use of Hessian
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
- Adapts to the data to make efficient use of Hessian
- Examines data to choose number/size of blocks
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
- Adapts to the data to make efficient use of Hessian
- Examines data to choose number/size of blocks
- Solves each block using NNLS
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
- Adapts to the data to make efficient use of Hessian
- Examines data to choose number/size of blocks
- Solves each block using NNLS
- Globally reallocates probability among blocks, calling itself recursively
Adaptive CNM

- Uses a divide and conquer approach
- Breaks the support set up into blocks
- Adapts to the data to make efficient use of Hessian
- Examines data to choose number/size of blocks
- Solves each block using NNLS
- Globally reallocates probability among blocks, calling itself recursively
- Guaranteed convergence to the solution
The HALT Study

ACNM Algorithm

Summing Up
Conclusions

- Where Interval Censoring is present in survival data, it can be allowed for in the analysis.
- The NPMLE Survival Function combined with Bootstrap methods can create an informative picture of survival progression in such cases.
- The ACNM algorithm provides a fast and robust solution to this problem.
Thanks to:

- My supervisor, Dr Yong Wang
- Andrew Jull and Varsha Parag of CTRU for providing the HALT data