

---

# **Development and application of methods in parametric survival models: interval censoring, inverse probability weighting and multistate survival models**

---

Thesis submitted for the degree of

Doctor of Philosophy

at the University of Leicester

by

Micki Hill, BSc. MSc.

Department of Health Sciences

University of Leicester

October 2022

---

# **Abstract**

---

## **Development and application of methods in parametric survival models: interval censoring, inverse probability weighting and multistate survival models**

**Micki Hill**

Although semi- and non-parametric approaches are frequently used to analyse survival data, there are advantages to using parametric survival models. This thesis develops and applies methods in parametric models to address, or investigate the impact of, issues that arise in survival data. Specifically, the thesis focuses on three key topics: interval censoring, inverse probability (IP) weighting and multistate models.

Data are interval-censored if the event is only known to occur within an interval. Often a single time (beginning, midpoint or end of the interval) is imputed and standard methods for right-censored data are employed. The impact of this naive imputation on interval-censored data is explored in a literature review and simulation study. As with all projects, the methods are demonstrated on an example dataset.

IP weighting can be used to estimate causal effects in the presence of confounding. Two types of weights, stabilised and unstabilised, can be employed and can result in different estimates when used in an IP weighted analysis on survival data. A simulation study was performed to confirm that both weights result in an unbiased estimator. A novel, closed-form variance estimator was then proposed for IP weighted parametric models, using M-estimation to account for the uncertainty in the weight estimation. The novel estimator was validated in a simulation study and can be used as an alternative to bootstrapping in large samples, especially when reproducibility or computational time are key concerns. User-friendly software was developed and made freely available.

Disease pathways may consist of multiple stages and warrant the use of multistate models. While modelling the transitions is straightforward, obtaining the predictions can be complex. Predictions were obtained for hospital acquired infection data from a flexible multistate model using a recently proposed, general simulation algorithm. Non-parametric estimates can serve as a reference and software was developed to facilitate this.

---

## Acknowledgements

---

I would like to start by thanking my supervisory team, Professor Paul Lambert and Dr Michael Crowther, without whom this achievement would not be possible. I have learnt so much over the last few years from them, whether it be programming, methodology or about the wider world of academia and publishing. In particular, I would like thank Paul for his patience, calmness and practicality, especially in the long months of writing up.

I would also like to thank the members of the Biostatistics group for their guidance and advice. A big thank you goes to the health sciences past and present PhD students, who have provided invaluable support, technical help and entertainment. Working from home has been a challenge for many of us and the strong network of students has been a blessing, whether it be to ask questions, to empathise with or just to have a laugh. Of course, I would also like to thank the board games regulars, who returned each fortnight despite my slight competitive streak.

My largest thanks goes to the people who have kept me sane over the last three years. To Katrina, Tasha and Pete – your friendship has provided some much-needed relief and I can always rely on you to bring a smile to my face. To my partner Sam, you have awaited this milestone as much as me, if nothing other than to have some new topics of conversation. You have been there for me throughout the process, through the highs and lows and for that, and all that you do, I am so thankful. Finally, to my parents Jo and Steve, words cannot describe how eternally grateful I am for your support and love over the years. You have always been there to motivate me, encourage me and inspire me – and this is no less true for the PhD. This achievement is as much yours as it is mine.

---

# Contents

---

<b>Abstract</b>	<b>I</b>
<b>Acknowledgements</b>	<b>III</b>
<b>Table of Contents</b>	<b>IV</b>
<b>List of Figures</b>	<b>XVIII</b>
<b>List of Tables</b>	<b>XXI</b>
<b>List of Abbreviations</b>	<b>XXII</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Context . . . . .	1
1.1.1 Survival Analysis . . . . .	1
1.1.2 Interval Censoring . . . . .	2
1.1.3 Inverse Probability Weighting . . . . .	4
1.1.4 Multistate Survival Models . . . . .	5
1.2 Thesis Aims . . . . .	6
1.3 Data Sources . . . . .	7
1.3.1 AIDS Clinical Trial Group Study 175 (ACTG175) Dataset .	8
1.3.2 Breast Cosmesis Dataset . . . . .	8
1.3.3 Hospital Acquired Infection (HAI) Dataset . . . . .	8
1.3.4 Right Heart Catheterisation (RHC) Dataset . . . . .	9
1.3.5 Sexually Transmitted Diseases (STD) Dataset . . . . .	9
1.4 Thesis Structure . . . . .	9
<b>2 Introduction to Methods</b>	<b>12</b>
2.1 Outline . . . . .	12
2.2 Introduction to Survival Analysis . . . . .	12
2.2.1 Definitions . . . . .	12
2.2.2 Relationships Between the Metrics . . . . .	13
2.2.3 Estimands for Survival Data . . . . .	14
2.3 Parametric Survival Models . . . . .	15

2.3.1	Notation, Censoring and Left Truncation . . . . .	15
2.3.2	Likelihood Function . . . . .	16
2.3.3	Parametric Proportional Hazards Models . . . . .	18
2.3.4	Royston-Parmar Models . . . . .	21
2.3.5	Accelerated Failure Time Models . . . . .	23
2.4	Other Analysis Approaches to Survival Data . . . . .	26
2.4.1	Kaplan-Meier Estimator . . . . .	26
2.4.2	Nelson-Aalen Estimator . . . . .	28
2.4.3	Cox Proportional Hazards Model . . . . .	29
2.5	Interval Censoring . . . . .	31
2.5.1	Definitions and Notation . . . . .	31
2.5.2	Review of Analysis Methods . . . . .	32
2.5.3	Naive Imputation . . . . .	33
2.5.4	Maximum Likelihood Estimation . . . . .	34
2.6	Inverse Probability Weighting . . . . .	35
2.6.1	Definitions and Notation . . . . .	35
2.6.2	Contrasts in Marginal Estimands . . . . .	36
2.6.3	Assumptions . . . . .	38
2.6.4	Inverse Probability Weighted Analysis Algorithm . . . . .	40
2.6.5	Treatment Model: Modelling the Propensity Score . . . . .	41
2.6.6	Weights . . . . .	43
2.6.7	Outcome Model: Inverse Probability Weighted Survival Model	45
2.7	Multistate Survival Models . . . . .	46
2.7.1	Definitions, Estimands and the Markov Assumption . . . . .	46
2.7.2	Modelling the Transitions . . . . .	48
2.7.3	Obtaining Predictions . . . . .	50
2.8	Summary . . . . .	51
<b>3</b>	<b>The Impact of Naive Imputation on Interval-censored Survival Data</b>	<b>52</b>
3.1	Outline . . . . .	52
3.2	Introduction . . . . .	52
3.3	Literature Review . . . . .	55
3.4	Motivating Dataset . . . . .	62
3.5	Simulation Study Methods . . . . .	64
3.5.1	Aims . . . . .	64
3.5.2	Data Generating Mechanisms . . . . .	64
3.5.3	Estimands . . . . .	68
3.5.4	Methods . . . . .	68
3.5.5	Performance Measures . . . . .	69

3.5.6	Software . . . . .	71
3.6	Simulation Study Results . . . . .	72
3.6.1	Exploratory Analysis . . . . .	72
3.6.2	Main Analysis . . . . .	74
3.7	Discussion . . . . .	84
3.8	Conclusion . . . . .	88
<b>4</b>	<b>Stabilised Versus Unstabilised Weights in an Inverse Probability Weighted Survival Analysis</b>	<b>89</b>
4.1	Outline . . . . .	89
4.2	Introduction . . . . .	89
4.3	Illustrative Dataset . . . . .	93
4.3.1	Description . . . . .	93
4.3.2	Methods . . . . .	93
4.3.3	Results . . . . .	94
4.3.4	Discussion . . . . .	97
4.4	Equivalence of IP Weighted Kaplan-Meier Estimators . . . . .	98
4.4.1	Standard Kaplan-Meier Estimator . . . . .	98
4.4.2	Weighted Kaplan-Meier Estimator . . . . .	98
4.5	Simulation Study Methods . . . . .	100
4.5.1	Aims . . . . .	100
4.5.2	Data Generating Mechanism . . . . .	100
4.5.3	Estimands . . . . .	103
4.5.4	Methods . . . . .	104
4.5.5	Performance Outcomes . . . . .	105
4.5.6	Software . . . . .	107
4.6	Simulation Study Results . . . . .	108
4.6.1	Exploratory Analysis . . . . .	108
4.6.2	Main Analysis . . . . .	109
4.7	Discussion . . . . .	114
4.8	Conclusion . . . . .	117
<b>5</b>	<b>Closed-form Variance Estimator for Inverse Probability Weighted Parametric Survival Models</b>	<b>118</b>
5.1	Outline . . . . .	118
5.2	Introduction . . . . .	118
5.3	Review of Current Methods . . . . .	121
5.3.1	Naive Variance Estimator . . . . .	121
5.3.2	Robust Variance Estimator . . . . .	121
5.3.3	Bootstrap Variance Estimator . . . . .	122

5.3.4	Closed-form Variance Estimators . . . . .	122
5.4	M-estimation Variance Estimator for IP Weighted Parametric Survival Models . . . . .	123
5.4.1	Notation, Estimands and Assumptions . . . . .	123
5.4.2	Estimation of IP Weighted Parametric Survival Models . . . . .	124
5.4.3	M-estimation Framework for IP Weighted Parametric Survival Models . . . . .	125
5.4.4	General M-estimation Variance Estimator for IP Weighted Parametric Survival Models . . . . .	127
5.4.5	M-estimation Variance Estimator for an IP Weighted Royston-Parmar Model . . . . .	129
5.5	Simulation Study Methods . . . . .	131
5.5.1	Aims . . . . .	131
5.5.2	Data Generating Mechanism . . . . .	132
5.5.3	Estimands . . . . .	133
5.5.4	Methods . . . . .	133
5.5.5	Performance Measures . . . . .	133
5.5.6	Software . . . . .	135
5.6	Simulation Study Results . . . . .	135
5.6.1	Exploratory Analysis . . . . .	135
5.6.2	Main Analysis . . . . .	137
5.6.3	Exploratory Analysis for the Small Sample Size . . . . .	142
5.7	Illustrative Examples . . . . .	143
5.7.1	General Methods . . . . .	143
5.7.2	STD Dataset . . . . .	144
5.7.3	AIDS Clinical Trials Group Study 175 (ACTG175) Dataset . . . . .	144
5.7.4	Right Heart Catheterisation (RHC) Dataset . . . . .	147
5.8	Discussion . . . . .	148
5.9	Conclusion . . . . .	153
<b>6</b>	<b>Software Development for the Closed-form Variance Estimator for Inverse Probability Weighted Parametric Survival Models</b>	<b>155</b>
6.1	Outline . . . . .	155
6.2	Introduction . . . . .	155
6.3	<code>stipw</code> Algorithm . . . . .	157
6.4	<code>stipw</code> Syntax and Options . . . . .	160
6.4.1	Syntax . . . . .	160
6.4.2	<i>tmoptions</i> : Treatment/Exposure Model . . . . .	161
6.4.3	<i>tmoptions</i> & <i>options</i> : Maximisation . . . . .	162
6.4.4	<i>options</i> : Outcome Model - <code>streg</code> . . . . .	162

6.4.5	<i>options</i> : Outcome Model - <code>stpm2</code>	163
6.4.6	<i>options</i> : Variance Estimation	164
6.4.7	<i>options</i> : Advanced	164
6.4.8	<i>options</i> : Reporting	165
6.5	Example Code with <code>stipw</code>	166
6.5.1	STD Dataset	166
6.5.2	AIDS Clinical Trials Group Study 175 (ACTG175) Dataset	171
6.5.3	Right Heart Catheterization (RHC) Dataset	174
6.6	Discussion	175
6.7	Conclusion	178
<b>7</b>	<b>Multistate Model Application to the Hospital Acquired Infection Dataset and Corresponding Software Development</b>	<b>179</b>
7.1	Outline	179
7.2	Introduction	179
7.3	Methods	182
7.3.1	The Extended Illness-death Model for HAIs	182
7.3.2	Metrics of Interest	183
7.3.3	Analysis Approaches	184
7.3.4	Simulation Approach	185
7.3.5	Software	187
7.4	Aalen-Johansen Estimates (Software Development)	188
7.4.1	Introduction	188
7.4.2	Transition Probabilities	189
7.4.3	Standard Errors	190
7.4.4	Length of Stay	192
7.4.5	Other Extensions	193
7.4.6	Example Code	193
7.4.7	Discussion	194
7.5	Results	195
7.5.1	Data	195
7.5.2	Transition Rates	196
7.5.3	Transition Probabilities	197
7.5.4	Attributable Mortality and Population Attributable Fraction	200
7.5.5	Length of Stay	202
7.6	Discussion	204
7.7	Conclusion	206
<b>8</b>	<b>Discussion</b>	<b>207</b>
8.1	Introduction	207

8.2	Summary of Thesis . . . . .	207
8.3	Project Specific Recommendations and Potential Impact of Research	210
8.4	Strengths and General Recommendations . . . . .	213
8.5	Limitations . . . . .	216
8.6	Further Work . . . . .	219
8.7	Final Conclusions . . . . .	223
<b>Appendix A</b>	<b>Draft Manuscript for the Closed-form Variance Estimator for IP Weighted Parametric Survival Models</b>	<b>225</b>
<b>Appendix B</b>	<b>Manuscript for the Multistate Model Application to the HAI Dataset</b>	<b>226</b>
<b>Appendix C</b>	<b>Code for <code>stipw</code></b>	<b>227</b>
<b>Appendix D</b>	<b>Code for <code>msaj</code></b>	<b>251</b>
<b>Appendix E</b>	<b>Additional Material for Chapter 3</b>	<b>262</b>
E.1	Motivating Dataset . . . . .	263
E.2	Simulation Study: Additional Figures for the Survival Probability at 24, 36 and 48 Months . . . . .	264
E.2.1	Aim 1: Bias . . . . .	264
E.2.2	Aim 2: Relative Percentage Error in ModSE . . . . .	267
E.2.3	Aim 3: Coverage . . . . .	270
E.3	Simulation Study: Additional Tables . . . . .	273
E.3.1	Log Hazard Ratio . . . . .	273
E.3.2	Survival Probability at 12 and 48 Months . . . . .	276
<b>Appendix F</b>	<b>Additional Material for Chapter 4</b>	<b>279</b>
F.1	Simulation Study: Additional Tables . . . . .	280
<b>Appendix G</b>	<b>Additional Material for Chapter 5</b>	<b>282</b>
G.1	M-estimation Variance Estimator for IP Weighted Parametric Models	283
G.1.1	Exponential Model . . . . .	283
G.1.2	Weibull Model . . . . .	284
G.1.3	Gompertz Model . . . . .	285
G.1.4	Log-logistic Model . . . . .	286
G.1.5	Log-normal Model . . . . .	288
G.2	Simulation Study: Starting Seeds . . . . .	290
G.3	Simulation Study: Additional Figures . . . . .	291
G.3.1	Large Samples . . . . .	291
G.3.2	Small Samples . . . . .	293

G.4	Simulation Study: Additional Tables . . . . .	297
G.4.1	Large Samples . . . . .	297
G.4.2	Small Samples . . . . .	301
<b>Appendix H</b>	<b>Additional Material for Chapter 6</b>	<b>305</b>
H.1	Data Variable Definitions . . . . .	305
<b>Appendix I</b>	<b>Additional Material for Chapter 7</b>	<b>309</b>
I.1	Additional Figures . . . . .	310
<b>Bibliography</b>		<b>314</b>

---

## List of Figures

---

1.1	Illustration of an exactly observed event and left-, right- and interval-censored events . . . . .	3
1.2	Example of a breast cancer multistate survival model, replicated from Putter <i>et al</i> [1] . . . . .	6
3.1	Illustration of the three naive imputation approaches (beginning, midpoint and end imputation) . . . . .	53
3.2	Survival probability estimates for the breast cosmesis data using the appropriate likelihood-based approach for interval-censored data and the three naive imputation approaches for the adjuvant chemotherapy group (left) and radiotherapy alone group (right). Note that the midpoint imputation approach overlaps the interval-censored approach	63
3.3	Left panel: The hazard function in the control group from the data generating mechanism for the four $\gamma$ values, where $c_a = 60$ months and $S_0(60) = 0.25$ (the survival probability at 60 months is 25%). Right panel: The survival function in the control group from the data generating mechanism for the four $\gamma$ values, where $S_0(60) = 0.25$ is shown by the solid lines and $S_0(60) = 0.75$ is shown by the dashed lines	67
3.4	Nested loop plot showing the bias of the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. For example, the fifth step represents a log hazard ratio of 0.92, sample size of 100, survival probability at 60 months of 25%, $\gamma$ value of 0.7 and interval width of 6 months. Note that the IC method often overlaps the exact method . . . . .	75
3.5	Nested loop plot showing the bias of the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	76
3.6	Nested loop plot showing the MISE (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	78
3.7	Nested loop plot showing the relative percentage error in model-based standard errors for the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	79

3.8	Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	80
3.9	Nested loop plot showing the coverage of the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	82
3.10	Nested loop plot showing the coverage of the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	83
4.1	Kaplan-Meier curve of time to reinfection in the raw (left) and weighted (right) data by race for the STD data. Stabilised and unstabilised weights gave exactly the same weighted graph . . . . .	96
4.2	Directed acyclic graph representing the data generating mechanism, based on Figure 1 from Hajage <i>et al</i> [2] . . . . .	101
4.3	Simulation study results showing bias for the marginal log hazard ratio where censoring was administrative only. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, Cox:US Semi-parametric (Cox) unstabilised, Cox:S Semi-parametric (Cox) stabilised . . . . .	110
4.4	Simulation study results showing bias for the marginal log hazard ratio where censoring was both administrative and intermittent. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, Cox:US Semi-parametric (Cox) unstabilised, Cox:S Semi-parametric (Cox) stabilised . . . . .	111
4.5	Simulation study results showing bias for the difference in marginal RMST where censoring was administrative only. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, NP:US Non-parametric (Kaplan-Meier) unstabilised, NP:S Non-parametric (Kaplan-Meier) stabilised. The red line (Kaplan-Meier estimator, unstabilised weights) cannot be seen as it is completely overlaid by the corresponding stabilised estimates. . . . .	112

4.6 Simulation study results showing bias for the difference in marginal RMST where censoring was both administrative and intermittent. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, NP:US Non-parametric (Kaplan-Meier) unstabilised, NP:S Non-parametric (Kaplan-Meier) stabilised. The red line (Kaplan-Meier estimator, unstabilised weights) cannot be seen as it is completely overlaid by the corresponding stabilised estimates	113
5.1 Univariate plot showing the spread of the model-based standard errors for the marginal log hazard ratio when $\gamma = 0.7$ , the treatment prevalence $\pi_Z = 0.1$ , the marginal hazard ratio $\exp(\beta) = 0.5$ and there is no intermittent censoring for the large sample size $n_{obs} = 10000$ . U:R, U:B and U:M are the robust, bootstrap and M-estimation variance estimators with unstabilised weights, respectively. S:R, S:B and S:M are the corresponding variance estimators with stabilised weights. The vertical red lines represent the empirical standard error for each weight, see Section 5.5.5. The value is given on the right-hand side in red text. The vertical yellow lines represent the average model-based standard error for each method. The value is given on the right-hand side in black text . . . . .	136
5.2 Univariate plot showing the spread of the model-based standard errors for the marginal log hazard ratio when $\gamma = 0.7$ , the treatment prevalence $\pi_Z = 0.5$ , the marginal hazard ratio $\exp(\beta) = 0.5$ and there is no intermittent censoring for the large sample size $n_{obs} = 10000$ . U:R, U:B and U:M are the robust, bootstrap and M-estimation variance estimators with unstabilised weights, respectively. S:R, S:B and S:M are the corresponding variance estimators with stabilised weights. The vertical red lines represent the empirical standard error for each weight, see Section 5.5.5. The value is given on the right-hand side in red text. The vertical yellow lines represent the average model-based standard error for each method. The value is given on the right-hand side in black text . . . . .	137
5.3 The relative percentage error of the stabilised variance estimators for the marginal log hazard ratio for the large sample size $n_{obs} = 10000$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	138

5.4	The relative percentage error of the stabilised variance estimators for the difference in marginal RMST for the large sample size $n_{obs} = 10000$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	139
5.5	The relative percentage error of the M-estimation variance estimators for the marginal log hazard ratio for the large sample size $n_{obs} = 10000$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	140
5.6	The relative percentage error of the M-estimation variance estimators for the difference in marginal RMST for the large sample size $n_{obs} = 10000$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	141
6.1	Output for <b>stipw</b> used on the STD dataset with stabilised weights for a Weibull model . . . . .	168
6.2	Output for <b>stipw</b> used on the STD dataset with stabilised weights for a Royston-Parmar model with 1 degree of freedom and the <b>noorthog</b> option (equivalent to a Weibull model fitted with <b>streg</b> ) . . . . .	169
6.3	Output for <b>stipw</b> used on the STD dataset with unstabilised weights for a Royston-Parmar model with 1 degree of freedom and the <b>noorthog</b> option (equivalent to a Weibull model fitted with <b>streg</b> ) . . . . .	170
6.4	Output for <b>stipw</b> used on the ACTG175 dataset with stabilised weights for a Royston-Parmar model with 2 degrees of freedom where treatment is a time-dependent effect with 1 degree of freedom . . . . .	172

6.5	The marginal hazard ratio for the ACTG175 dataset as a function of time with confidence intervals (based on the M-estimation variance estimates) . . . . .	173
7.1	Extended illness-death model for discharge and death with and without a hospital acquired infection (HAI) . . . . .	182
7.2	Transition rates from the “AIC” model with 95% confidence intervals (shaded region) for the HAI data. Point estimates and confidence intervals were defined from the time of the first event until the last event for each transition (solid lines). The point estimates were extrapolated to cover the interval [0, 82] (dashed line) . . . . .	197
7.3	Transition probabilities from state 1 at time 0 to each state for the “AIC” model with 95% confidence intervals for the HAI data . . . . .	198
7.4	Transition probabilities from state 1 at time 0 to each state for the different approaches for the HAI data. Note that there is considerable overlap between the “RP(4)” and “AIC” estimates . . . . .	199
7.5	Transition probabilities from state 2 at time 3 to the relevant states for the different approaches for the HAI data . . . . .	200
7.6	Attributable mortality (AM) and population attributable fraction (PAF) of HAIs for the different approaches for the HAI data . . . . .	201
7.7	Length of stay in hospital without (state 1, left) and with (state 2, right) a HAI from state 1 at time 0 for the different approaches for the HAI data . . . . .	202
7.8	Total length of stay in hospital from state 1 at time 0 for the “AIC” model with 95% confidence intervals for the HAI data . . . . .	203
7.9	Residual length of stay in hospital with a HAI (state 2), from state 2 at time 3 for the HAI data. In the left panel, for the different approaches. In the right panel, for the “AIC” model with 95% confidence intervals . . . . .	204
E.1	Survival probability estimates and confidence intervals for the breast cosmesis data using the appropriate likelihood-based approach (top left), beginning imputation (top right), midpoint imputation (bottom left) and end imputation (bottom right) . . . . .	263
E.2	Nested loop plot showing the bias of the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	264
E.3	Nested loop plot showing the bias of the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	265

E.4 Nested loop plot showing the bias of the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	266
E.5 Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	267
E.6 Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	268
E.7 Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	269
E.8 Nested loop plot showing the coverage of the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	270
E.9 Nested loop plot showing the coverage of the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	271
E.10 Nested loop plot showing the coverage of the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method . . . . .	272

G.1	The relative percentage error of the unstabilised variance estimators for the marginal log hazard ratio for the large sample size $n_{obs} = 10000$ . The unstabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	291
G.2	The relative percentage error of the unstabilised variance estimators for the difference in marginal RMST for the large sample size $n_{obs} = 10000$ . The unstabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	292
G.3	The relative percentage error of the stabilised variance estimators for the marginal log hazard ratio for the small sample size $n_{obs} = 200$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	293
G.4	The relative percentage error of the stabilised variance estimators for the difference in marginal RMST for the small sample size $n_{obs} = 200$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	294

G.5 The relative percentage error of the M-estimation variance estimators for the marginal log hazard ratio for the small sample size $n_{obs} = 200$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	295
G.6 The relative percentage error of the M-estimation variance estimators for the difference in marginal RMST for the small sample size $n_{obs} = 200$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent $\gamma = 0.7$ and the bottom panels represent $\gamma = 1.4$ . The first, second and third column show treatment prevalence $\pi_Z = 0.1, 0.25$ and $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio $\exp(\beta) = 0.5$ and the second two scenarios represent a marginal hazard ratio $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC) . . . . .	296
I.1 Transition rates for the different approaches for the HAI data. The Epanechnikov kernel was used for smoothed non-parametric estimates. Estimates from the parametric approaches were defined from the time of the first event until the last event for each transition by a solid line and were extrapolated to cover the interval $[0, 82]$ by a dashed line. The smoothed non-parametric estimates were truncated up to 8 days before the last event, as it was not believed that the transition rates increased so drastically at the time of the last event . . . . .	310
I.2 Transition probabilities from state 2 at time 3 to the relevant states for the “AIC” model with 95% confidence intervals for the HAI data . . . . .	311
I.3 Attributable mortality (AM) and population attributable fraction (PAF) of HAIs for the “AIC” model with 95% confidence intervals for the HAI data . . . . .	312
I.4 Length of stay in hospital without (state 1, left panel) and with (state 2, right panel) a HAI starting from state 1 at time 0 for the “AIC” model with 95% confidence intervals for the HAI data . . . . .	313

---

## List of Tables

---

2.1	Contribution from individual $i$ to the likelihood function for different types of censoring (an extension of Table 1.1 from [3] using Equation 1.13 from [3]) . . . . .	34
3.1	Factors that were varied or investigated in each simulation study, denoted with a ✓ . . . . .	59
3.2	Parameter estimates from a Weibull model fitted to the breast cosmesis data with the appropriate likelihood-based method for interval-censored data compared to the three naive imputation methods . . . . .	62
3.3	$\lambda$ values for the data generating mechanism, rounded to 3 significant figures (the exact values were used in the simulation study) . . . . .	67
3.4	Description of simulation study notation, a subset of Table 2 from Morris <i>et al</i> [4] . . . . .	69
3.5	Estimated maximum variance for each of the first two estimands from the preliminary simulation with 100 iterations, corresponding $n_{sim}$ required so that the MCSE for bias is less than or equal 0.005 and the estimated maximum MCSE when $n_{sim} = 9200$ . . . . .	72
4.1	Baseline data and standardised differences of the raw and weighted data (using stabilised weights) in the STD dataset . . . . .	95
4.2	Estimates of the marginal hazard ratio and difference in marginal RMST at 4 years for the different methods and types of weights on the STD data . . . . .	97
4.3	Parameters used in the data generating mechanism (treatment model and covariances), based on those from Hajage <i>et al</i> [2], Table 1 . . . . .	103
4.4	True values of the marginal hazard ratio (HR), RMST in the treatment group at time 20 ( $RMST_1(20)$ ) and the difference in marginal RMST at time 20 ( $\Delta_\mu(20)$ ) in the data generating mechanism . . . . .	104
4.5	$n_{obs} = 2000$ : Estimated maximum variance for each estimand from the preliminary simulation with 1000 iterations, corresponding $n_{sim}$ required so that the MCSE for bias is less than or equal the acceptable value and the estimated maximum MCSE when $n_{sim} = 2500$ . . . . .	106
4.6	$n_{obs} = 10000$ : Estimated maximum variance for each estimand from the preliminary simulation with 1000 iterations, corresponding $n_{sim}$ required so that the MCSE for bias is less than or equal the acceptable value and the estimated maximum MCSE when $n_{sim} = 1000$ . . . . .	106
4.7	Starting seeds and computational time for the preliminary and main simulation studies . . . . .	107

5.1	Estimated $n_{sim}$ required so that the MCSE for the relative percentage error in the model-based standard error is less than or equal 1 percentage point and the estimated maximum MCSE when $n_{sim} = 7700$ (for the 10000 sample size) . . . . .	135
5.2	Computational time (hours) to calculate the marginal log hazard ratio and difference in marginal RMST for both sample sizes for both weights in the simulation study. The table gives summary data across the 24 scenarios (batches) . . . . .	142
5.3	Standard errors for the marginal hazard ratio and difference in marginal RMST at 4 years for the different variance estimators and types of weights for the STD dataset . . . . .	144
5.4	Computational time (seconds) to calculate the difference in marginal RMST for the different variance estimators and for the different datasets. Results are for stabilised weights . . . . .	145
5.5	Standard errors for the difference in marginal RMST at 3 years for the different variance estimators and types of weights for the ACTG175 dataset . . . . .	146
5.6	Standard errors for the difference in marginal RMST at 5 years for the different variance estimators and types of weights for the RHC dataset . . . . .	148
7.1	Summary of the key features of the <code>mstate</code> package (in R), <code>etm</code> package (in R) and <code>msaj</code> command (part of the <code>multistate</code> package in Stata) for calculating Aalen-Johansen estimates . . . . .	189
7.2	AIC for each parametric model fitted to each transition separately (to determine the “AIC” model) on the HAI dataset . . . . .	196
E.1	Bias with MCSE for the log hazard ratio for sample size 100 and interval width 12 months . . . . .	273
E.2	Relative percentage error of the model-based standard errors with MCSE for the log hazard ratio for sample size 100 and interval width 12 months . . . . .	274
E.3	Coverage with MCSE for the log hazard ratio for sample size 100 and interval width 12 months . . . . .	275
E.4	Bias with MCSE for the survival probability at 12 and 48 months (in the control group) for $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100 . . . . .	276
E.5	Relative percentage error of the model-based standard errors with MCSE for the survival probability at 12 and 48 months (in the control group) for $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100 . . . . .	277
E.6	Coverage with MCSE for the survival probability at 12 and 48 months (in the control group) for $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100 . . . . .	278
F.1	Bias (MCSE) of the marginal log hazard ratio $\beta$ . Note that $n_{sim} = 2500$ for $n_{obs} = 2000$ and $n_{sim} = 1000$ for $n_{obs} = 10000$ . . . . .	280
F.2	Bias (MCSE) of the difference in marginal RMST $\Delta_\mu(20)$ . Note that $n_{sim} = 2500$ for $n_{obs} = 2000$ and $n_{sim} = 1000$ for $n_{obs} = 10000$ . . . . .	281

G.1	Starting seeds for the preliminary and main simulation studies in Chapter 5 for each of the 24 batches . . . . .	290
G.2	Relative % error of ModSE (MCSE) for the marginal log hazard ratio for $\gamma = 0.7$ , $\lambda = 0.15$ and $n_{obs} = 10000$ . . . . .	297
G.3	Relative % error of ModSE (MCSE) for the marginal log hazard ratio for $\gamma = 1.4$ , $\lambda = 0.003$ and $n_{obs} = 10000$ . . . . .	298
G.4	Relative % error of ModSE (MCSE) for the difference in marginal RMST $\Delta_\mu(20)$ for $\gamma = 0.7$ , $\lambda = 0.15$ and $n_{obs} = 10000$ . . . . .	299
G.5	Relative % error of ModSE (MCSE) for the difference in marginal RMST $\Delta_\mu(20)$ for $\gamma = 1.4$ , $\lambda = 0.003$ and $n_{obs} = 10000$ . . . . .	300
G.6	Relative % error of ModSE (MCSE) for the marginal log hazard ratio for $\gamma = 0.7$ , $\lambda = 0.15$ and $n_{obs} = 200$ . . . . .	301
G.7	Relative % error of ModSE (MCSE) for the marginal log hazard ratio for $\gamma = 1.4$ , $\lambda = 0.003$ and $n_{obs} = 200$ . . . . .	302
G.8	Relative % error of ModSE (MCSE) for the difference in marginal RMST $\Delta_\mu(20)$ for $\gamma = 0.7$ , $\lambda = 0.15$ and $n_{obs} = 200$ . . . . .	303
G.9	Relative % error of ModSE (MCSE) for the difference in marginal RMST $\Delta_\mu(20)$ for $\gamma = 1.4$ , $\lambda = 0.003$ and $n_{obs} = 200$ . . . . .	304
H.1	STD dataset variable description, as described in R package <b>KMsurv</b> [5] . . . . .	306
H.2	ACTG175 dataset variable description, as described in R package <b>speff2trial</b> [6] . . . . .	307
H.3	ACTG175 dataset variable description, as described in R package <b>speff2trial</b> [6] (continued) . . . . .	308

---

## List of Abbreviations

---

**ACTG175:** AIDS Clinical Trial Group Study 175

**ADEMP:** Aims, Data generating mechanisms, Estimands, Methods, Performance measures

**AFT:** Accelerated Failure Time

**AIC:** Akaike Information Criterion

**AJ:** Aalen-Johansen

**AM:** Attributable Mortality

**BIC:** Bayesian Information Criterion

**DF:** Degrees of Freedom

**EM:** Expectation-Maximisation

**EmpSE:** Empirical Standard Error

**HAI:** Hospital Acquired Infection

**HR:** Hazard Ratio

**IC:** Interval Censoring

**ICM:** Iterative Convex Minorant

**i.i.d.:** Independent and Identically Distributed

**IP:** Inverse Probability

**KM:** Kaplan-Meier

**MCSE:** Monte Carlo Standard Error

**MISE:** Mean Integrated Squared Error

**MLE:** Maximum Likelihood Estimate

**ModSE:** Model-based Standard Error

**MST:** Mean Survival Time

**PAF:** Population Attributable Fraction

**RHC:** Right Heart Catheterisation

**RMST:** Restricted Mean Survival Time

**RP:** Royston-Parmar

**SE:** Standard Error

**STD:** Sexually Transmitted Disease(s)

**SUTVA:** Stable Unit Treatment Value Assumption

# Chapter 1

---

## Introduction

---

### 1.1 Context

#### 1.1.1 Survival Analysis

The time it takes for an event to occur after a specified time origin is termed time-to-event data (or survival data if the event of interest is death). An example is the time it takes for an individual to develop a local recurrence after undergoing surgery for breast cancer. A key feature of time-to-event data is censoring. If the exact time an event occurs is known, for example, the date of death, the event time is said to be observed exactly. Note that the starting time needs to be well defined. Alternatively, an event time is said to be censored if the time at which the event occurs is not observed. The most common type of censoring is right censoring and arises when the event time is greater than the last observed time. For example, individuals who are event-free at the end of a study will be censored at the last assessment time (known as administrative censoring). Survival analysis is used to analyse such data and appropriately incorporate the censored event times. A number of useful metrics can then be estimated, for example, the median event time, the probability the event occurs by a specified time and/or a treatment effect size.

There are three main analysis approaches to time-to-event data: non-parametric, semi-parametric and parametric methods. The most common non-parametric esti-

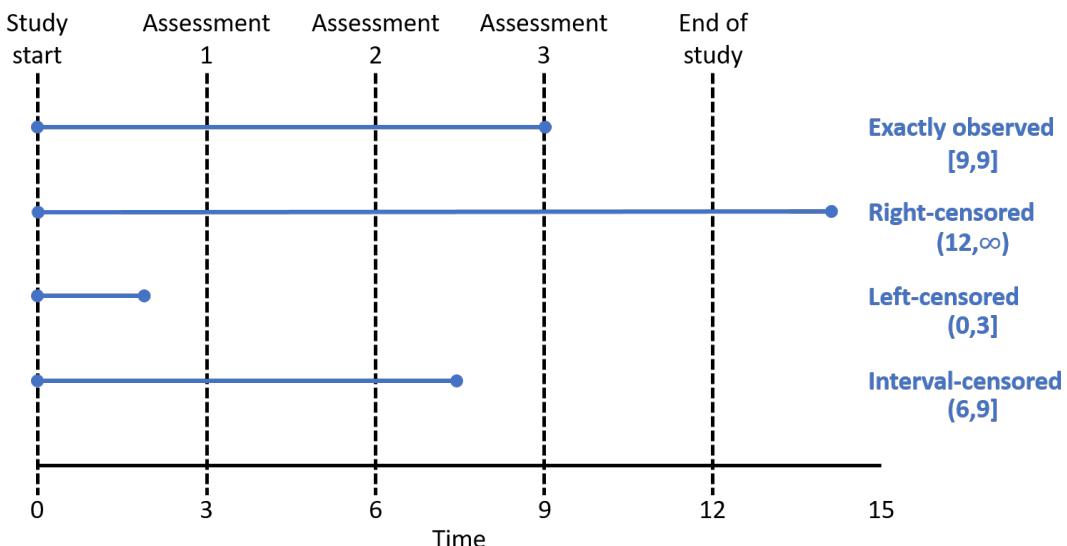
mator for time-to-event data is the Kaplan-Meier estimator. It imposes no assumptions on the distribution of event times nor assumes a specific relationship between covariates and the event time [7]; however, only categorical covariates are supported. The most common semi-parametric method is the Cox proportional hazards model. Continuous covariates can be incorporated and the approach makes no assumptions on the shape of the baseline hazard function. Although well suited to estimate the treatment effect when the hazards are proportional, the unknown baseline hazard function makes estimating other metrics, such as the probability of survival, more difficult.

Parametric models are less frequently used due to the assumptions required on the survival times and relationship with the covariates. To address this, a relatively new, flexible class of parametric models has been proposed that can allow for complex hazard functions. Royston-Parmar (RP) models [8] utilise restricted cubic splines to model the effect of time on the log cumulative hazard scale. Covariates can be modelled with time-dependent effects, offering increased flexibility and allowing for non-proportional hazards. As the baseline hazard function is known, a range of useful estimands (see Section 2.2.3) can easily be estimated. As such, parametric models have great potential and form the focus of the thesis.

This thesis builds on three main areas in survival analysis: interval censoring, inverse probability (IP) weighting and multistate survival models. These topics are introduced in the following subsections.

### 1.1.2 Interval Censoring

In addition to right censoring, events can be left- or interval-censored. Left censoring occurs when an individual has experienced the event before the first observed time, for example, before their first post-baseline assessment. Interval censoring arises when the event of interest occurs between two observed times, for example, between two assessments in a clinical trial. An example of interval censoring is in cancer clinical trials, where assessments may be taken every few months to identify evidence of disease progression. Alternatively, for an observational study in dentistry, check-



**Figure 1.1:** Illustration of an exactly observed event and left-, right- and interval-censored events

ups may be conducted annually to identify the emergence of a new tooth. Figure 1.1 considers a hypothetical clinical trial with five assessments (one at study entry, three intermediate assessments and one at the end of study). Exactly observed and right-, left- and interval-censored event times are demonstrated.

Data are frequently interval-censored, although are rarely analysed as such. Interval-censored data can be appropriately analysed using likelihood-based methods [3]. Historically, a lack of available software has hindered the use of appropriate methods; however, this is no longer the case. Despite this, often a single event time is imputed and the data analysed as if they were exactly observed/right-censored. A common approach, especially in oncology clinical trials, is to take the assessment date when disease progression was confirmed to be the event time (end imputation). Other single imputation approaches include using the first assessment date (beginning imputation, not often performed) or the midpoint between the two assessments (midpoint imputation). Although naive imputation methods are frequently used for interval-censored data, the question of whether this is appropriate remains. This forms the first aim of the thesis.

### 1.1.3 Inverse Probability Weighting

Randomised clinical trials allow us to answer questions surrounding causality; however, it may not always be ethical, feasible or timely to perform such a study. Substantial research has therefore focused on how to obtain causal estimates from observational studies in the presence of confounders, which are variables that are associated with both the treatment (or exposure) and outcome. One group of methods use the propensity score to reduce confounding and estimate a marginal treatment effect. The propensity score is the probability of being assigned the treatment, conditional on the measured baseline confounding variables. Four propensity score methods are commonly cited in the literature: matching [9, 10], stratification [11], regression adjustment [9] and weighting [12], see also further references [13–15].

The thesis will focus on IP weighting, which uses the inverse of the propensity score as weights to obtain a pseudo-population where the distribution of the measured baseline confounders is independent of the treatment assignment [14]. This means that individuals with under-represented covariate patterns in their treatment group are given a greater weight. If certain assumptions hold, two types of weights, stabilised and unstabilised, are commonly used to estimate a marginal treatment effect in the sampled population. Unstabilised weights are defined as the inverse of the probability of receiving the assigned treatment, given the measured confounders. Stabilised weights multiply the unstabilised weights by the unconditional probability of receiving the assigned treatment. Other weighting strategies are available that target different populations, for example, the average treatment effect in the treated [16]. A worked example of IP weighting is given by Austin [17] on the effect of in-hospital smoking cessation counselling on mortality.

When performing an IP weighted analysis on survival data, stabilised and unstabilised weights can result in slightly different point estimates. In the context of time-varying or continuous treatments, stabilised weights have been recommended, as they can produce less extreme weights and typically can reduce the variance [15]. In terms of point estimation, both weights result in an unbiased estimator [18]. The second aim of the thesis is to confirm this in the setting of a fixed, binary treatment.

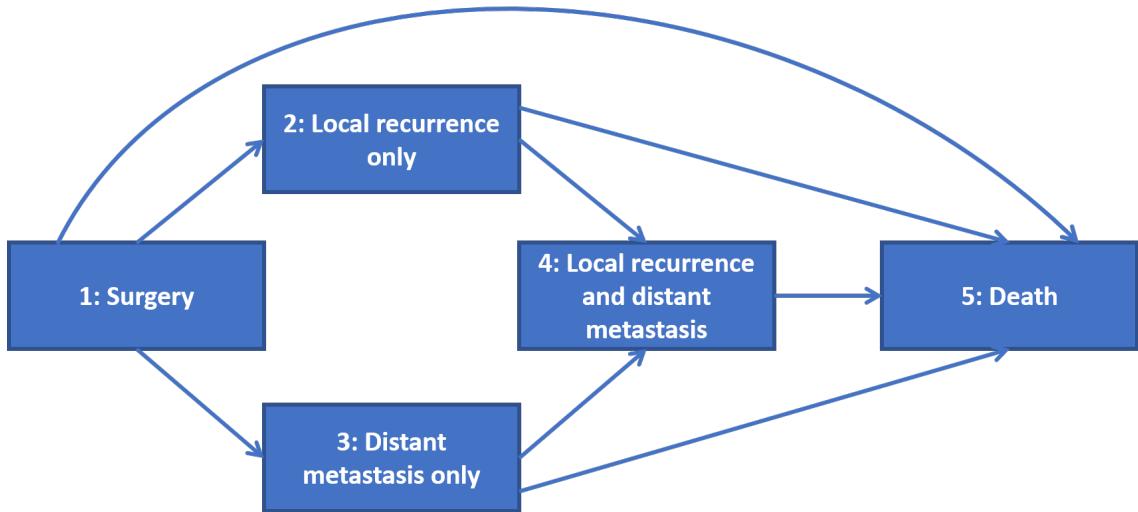
Correct variance estimation is necessary for valid inference. Historically, a robust variance estimator has been used for IP weighted analyses; however, this has been shown to be conservative [19]. Instead, bootstrapping has been recommended [19]. However, bootstrapping can be computationally intensive and requires knowledge of the random number generator to reproduce the results. Austin [19] commented that a closed-form variance estimator for survival data would be useful. Previous work has focused on the Cox proportional hazards model [2, 20, 21], while IP weighted *parametric* survival models are yet to receive the same consideration. This observation motivates the third aim of the thesis.

#### 1.1.4 Multistate Survival Models

Disease pathways may be complex and consist of intermediate, competing and/or subsequent events of interest, as well as the primary event. One example is the breast cancer multistate model described by Putter *et al* [1]. Following surgery for breast cancer, individuals were at risk of developing either a distant metastases or local recurrence. Once either event had occurred, the individual was then at risk of developing the other event as well. At any point, the individual was also at risk of dying. Figure 1.2, replicated from Putter *et al* [1], illustrates this multistate model example.

Multistate models can simultaneously model multiple events and transitions of interest, which may better represent the complex nature of a disease. Many useful metrics can be obtained, including the rate at which transitions between events are made, the probability of being in each state at a specified time point and the expected length of time spent in a state. With an increasing number of large registry-based data sources becoming available, there is greater opportunity for multistate models to improve the understanding of disease processes [22]. The potential of multistate models are also evident in health economics, where the length of time spent in states can be translated into costs.

The transitions in a multistate model can be modelled relatively easily; however, obtaining the corresponding predictions can be difficult. Various approaches



**Figure 1.2:** Example of a breast cancer multistate survival model, replicated from Putter *et al* [1]

have been suggested, for example, von Cube *et al* [23] provide a tutorial where the transition rates were assumed to be constant and predictions were obtained analytically. However, assuming constant transition rates may not always be reasonable. Crowther and Lambert [22] have recently proposed a more flexible approach, where a general simulation algorithm is used to obtain predictions. A demonstration of how predictions can be calculated using this more flexible approach would be useful. This forms the final aim of the thesis.

## 1.2 Thesis Aims

The overarching aim of the thesis is to develop and apply methods in parametric survival models to address, or investigate the impact of, issues that arise from both randomised clinical trials and observational data. In particular, the thesis focuses on interval censoring, IP weighting and multistate models, which have been introduced in Sections 1.1.2, 1.1.3 and 1.1.4, respectively. Each subsection introduced the topic and motivated the specific aims of the thesis, which are described in the following paragraphs.

The first aim of the thesis is to investigate the performance of naive imputation techniques on interval-censored survival data and to compare them to the likelihood-

based method that appropriately accounts for interval censoring. This will provide insight on when the performance of naive imputation is sub-optimal and an alternate approach should be employed.

The second aim is to confirm that both stabilised and unstabilised weights result in an unbiased estimator in an IP weighted survival analysis, with a fixed, binary treatment/exposure. This will provide a recommendation as to which IP weight should be used in this setting in terms of point estimation.

The third aim is to develop a closed-form variance estimator for IP weighted parametric survival models, which uses M-estimation to take into account the associated uncertainty in the weight estimation. The performance of the proposed estimator will be evaluated and corresponding software will be developed to facilitate implementation. This work will provide a computationally efficient and more easily reproducible alternative to bootstrapping, which may be especially useful when multiple analyses on large datasets (such as registry data) are required.

The final aim is to demonstrate how predictions can be obtained from a multistate model using the general simulation algorithm proposed by Crowther and Lambert [22]. This approach will be compared to a simplified analysis on the same data and existing software will be majorly redeveloped and extended to provide a reference method for the comparison. This application will provide a worked example for analysts who wish to analyse multistate models, without having to assume constant transition rates.

## 1.3 Data Sources

Due to the variety of topics investigated in the thesis, multiple datasets were used to demonstrate and motivate the methods described. All data sources are freely available and details on how to access them are given below, along with a brief description. Section 1.4 details where each dataset is used in the thesis.

### 1.3.1 AIDS Clinical Trial Group Study 175 (ACTG175)

#### Dataset

The AIDS Clinical Trial Group Study 175 (ACTG175) was a randomised clinical trial, originally reported by Hammer *et al* [24]. The study compared monotherapy with zidovudine or didanosine with combination therapy with zidovudine and didanosine or zidovudine and zalcitabine in adults infected with HIV type I, whose CD4 T cell counts were between 200 and 500 per cubic millimetre [6]. The outcome was years to the first occurrence of (i) a decline CD4 T cell count of at least 50, (ii) an event indicating progression to AIDS or (iii) death [6]. The ACTG175 dataset can be obtained from the `speff2trial` package in R (<https://CRAN.R-project.org/package=speff2trial>) [6] and consists of 2139 individuals and 17 covariates (excluding treatment).

### 1.3.2 Breast Cosmesis Dataset

The study aimed to compare the cosmetic effects of radiotherapy alone against radiotherapy with adjuvant chemotherapy and was initially reported by Beadle *et al* [25, 26]. The event of interest was time to the first appearance of moderate or severe breast retraction. The data can be accessed from <https://www.stata-press.com/data/r17/st.html> or obtained from numerous R packages (for example, `icensBKL` [27], `ICBayes` [28] and `interval` [29]) and consists of 94 individuals. Although other covariates were reported, only treatment was of interest.

### 1.3.3 Hospital Acquired Infection (HAI) Dataset

This is a sample from an observational cohort study conducted to analyse the burden of hospital acquired infections (HAIs) in intensive care, see Beyersmann *et al* [30] for details. There were six states in the multistate model: in hospital without a HAI, discharged without a HAI, dead without a HAI, in hospital with a HAI, discharged after having a HAI and dead after having a HAI. The `los.data` can be obtained from the `etm` package in R (<https://CRAN.R-project.org/package=etm>) [31] and

consists of 756 individuals. There were no covariates in this dataset.

### 1.3.4 Right Heart Catheterisation (RHC) Dataset

The observational study was performed to investigate the effectiveness of right heart catheterisation (RHC) in the initial care of critically ill patients and was initially reported by Connors *et al* [32]. The study compared patients receiving a RHC within 24 hours of admission to those who did not in hospitalised adult patients at five medical centres in the US. The outcome was time to death (in years). The `rhc` dataset can be obtained from the `Hmisc` package in R (<https://CRAN.R-project.org/package=Hmisc>) [33] and consists of 5735 individuals and 53 covariates (excluding RHC treatment).

### 1.3.5 Sexually Transmitted Diseases (STD) Dataset

The study was conducted to investigate the risk factors associated with reinfection for patients with sexually transmitted diseases (STDs); full details are given by Klein and Moeschberger [34]. The event of interest was time to reinfection. The `std` dataset was obtained from the `KMsurv` package in R (<https://CRAN.R-project.org/package=KMsurv>) [5] and consists of 877 individuals and 16 covariates.

## 1.4 Thesis Structure

Chapter 2 introduces the key concepts of survival analysis including definitions, relationships between the metrics and estimands. The focus of the thesis is on parametric survival models and, therefore, this chapter describes how time-to-event data can be modelled using proportional hazards models, Royston-Parmar models and accelerated failure time (AFT) models. The Kaplan-Meier (KM) estimator and Cox proportional hazards model are used as comparators in the thesis and are described in this chapter. The chapter continues on to define and describe the key modelling approaches for each of the three main topics of the thesis: interval censoring, IP weighting and multistate models.

The performance of naive imputation on interval-censored survival data is investigated in Chapter 3 with a literature review and comprehensive simulation study. The key findings of the review are summarised and areas where knowledge is lacking are highlighted. The simulation study addresses the gaps in the literature and aims to investigate which naive approach performs the best and in what scenarios it is particularly important to use appropriate methods for interval-censored data. The imputation approaches are illustrated on the breast cosmesis dataset.

Chapter 4 investigates whether stabilised or unstabilised weights should be used in an IP weighted analysis, when applied to survival data with a fixed, binary treatment/exposure. The chapter begins by illustrating an IP weighted analysis on the STD dataset, before conducting a simulation study to confirm that the two IP weighted estimators are unbiased. A proof is also given to show why an IP weighted Kaplan-Meier estimator gives equivalent results, regardless of which weight is used.

While Chapter 4 focuses on point estimation, Chapter 5 considers variance estimation in an IP weighted survival analysis. The chapter begins by reviewing existing variance estimators and then proposes a closed-form variance estimator for a range of IP weighted parametric survival models. Its performance is evaluated in a simulation study (an extension to the simulation study in Chapter 4) and the proposed variance estimator is demonstrated on the ACTG175, RHC and STD datasets. A manuscript of this work has been drafted and a copy is given in Appendix A.

The variance estimator in Chapter 5 can be implemented using the newly written, user-friendly **Stata** command **stipw**, produced as part of the work of this thesis. Chapter 6 reviews related software in **R** and **Stata**, before discussing the algorithm and syntax of **stipw**. Example code is given, which corresponds to the applications in Chapter 5. **stipw** can be accessed from the SSC archive or from GitHub at <https://github.com/Micki-Hill/stipw>. A copy of the code is given in Appendix C.

Chapter 7 extends the multistate analysis on the HAI dataset, described by von Cube *et al* [23], by using a general simulation algorithm to provide predictions. The two analysis approaches are compared against a non-parametric, ref-

erence method. This required an extension to the user-written **Stata** command `msaj`, part of the `multistate` package [22], and details of the software development are given in this chapter. `msaj` can be accessed on the SSC archive (through the `multistate` package) or from GitHub at <https://github.com/RedDoorAnalytics/multistate/tree/main/msaj>. A copy of the code is given in Appendix D. This work has been published [35] and a copy of the manuscript is provided in Appendix B.

Finally, Chapter 8 discusses the key results, strengths and limitations of the thesis. Recommendations based on this work and possible extensions are then proposed.

# Chapter 2

---

## Introduction to Methods

---

### 2.1 Outline

This chapter introduces survival analysis and, in particular, introduces the definitions, relationships between the metrics and estimands of interest. The chapter continues on to discuss parametric models (proportional hazards models, Royston-Parmar models and AFT models), non-parametric estimators (Kaplan-Meier estimator and Nelson-Aalen estimator) and the Cox proportional hazards model. The important features of the three main topics (interval censoring, IP weighting and multistate survival models) are then introduced. The definitions, assumptions and modelling approaches are discussed for each topic and form a foundation for the subsequent chapters of the thesis.

### 2.2 Introduction to Survival Analysis

#### 2.2.1 Definitions

Following Collett [36], let  $T$  denote the random variable associated with the event time, which can take any non-negative value. The cumulative density function of variable  $T$ ,  $F(t)$ , represents the probability that an individual experiences the event of interest by time  $t$ . This is also known as the cumulative incidence function and

is the integral of the probability density function,  $f(t)$ .

$$F(t) = \text{P}(T \leq t) = \int_0^t f(u)du$$

The survivor function,  $S(t)$ , is the probability that an individual is event-free beyond time  $t$ .

$$S(t) = \text{P}(T > t) = 1 - F(t) \quad (2.1)$$

Another important metric is the hazard function,  $h(t)$ , or hazard rate. It is the instantaneous failure rate at time  $t$ . Consider the conditional probability of the event occurring in the next time interval (between  $t$  and  $t + \delta t$ ), given that the individual had not experienced the event before time  $t$ . The hazard function is then the limit as  $\delta t$  goes to zero of the conditional probability divided by the interval  $\delta t$ , resulting in a rate.

$$h(t) = \lim_{\delta t \rightarrow 0} \left\{ \frac{\text{P}(t \leq T < t + \delta t | T \geq t)}{\delta t} \right\} \quad (2.2)$$

The cumulative hazard function,  $H(t)$ , is the cumulative rate of an event occurring by time  $t$  and is the integral of the hazard function.

$$H(t) = \int_0^t h(u)du$$

For a tutorial on survival analysis, please refer to Collett [36], Aalen *et al* [37] and Klein *et al* [38].

### 2.2.2 Relationships Between the Metrics

The definitions in Section 2.2.1 lead to some useful relationships between the different functions:

$$h(t) = \frac{f(t)}{S(t)} \quad (2.3)$$

$$h(t) = -\frac{d}{dt} [\log \{S(t)\}]$$

$$S(t) = \exp \{-H(t)\} = \exp \left\{ \int_0^t h(u)du \right\} \quad (2.4)$$

### 2.2.3 Estimands for Survival Data

A number of estimands can be estimated from time-to-event data including the survival probability and hazard rate at a certain time point, as defined in Section 2.2.1. The survival probability is a useful measure of absolute risk and is often represented graphically. If a single summary measure of the data is required (for example, to compare between methods or groups), it may be difficult to choose what value of  $t$  the survival probability should be evaluated at. The mean survival time (MST), denoted by  $\mu$ , has been suggested as a single summary measure and is defined as the integral of the survival function over the full domain of  $T$  (from  $t = 0$  to  $t = \infty$ ).

$$\mu = E(T) = \int_0^\infty S(t)dt \quad (2.5)$$

An alternative to the MST is the more commonly used restricted mean survival time (RMST) [39, 40], denoted as  $\mu(t_r)$ . The RMST integrates the survival curve from time  $t = 0$  to the time horizon  $t = t_r$ , avoiding extrapolating the estimated survival function to infinity.

$$\mu(t_r) = E \{ \min (T, t_r) \} = \int_0^{t_r} S(t)dt$$

The difference in RMST between two groups can be a useful measure of a covariate effect size [41]. Proportional hazards models and AFT models summarise the covariate effect with a hazard ratio (HR) and time ratio, respectively. However, when the necessary assumptions do not hold, the RMST can be a suitable alternative. The hazard ratio and time ratio are defined in Sections 2.3.3 and 2.3.5, respectively.

## 2.3 Parametric Survival Models

### 2.3.1 Notation, Censoring and Left Truncation

Censoring was introduced in Section 1.1.1 and occurs when the event time is unknown. Let  $C$  be the random variable denoting the censoring time and now let  $T^*$  be the random variable denoting the event time. Unless otherwise stated, let us consider right censoring. For individual  $i$ , let  $t_i^*$  be their event time and  $c_i$  their censoring time. The observed survival time for individual  $i$  is  $t_i = \min(t_i^*, c_i)$ . Henceforth,  $t_i^*$  will be referred to as the event time and  $t_i$  referred to as the survival time, which is the minimum of the event time and censoring time. Let  $\delta_i = 1(t_i^* \leq c_i)$  be the event indicator (1 if the event occurred, 0 otherwise), where  $1(\cdot)$  is the indicator function.

Let  $X = \{X_1, \dots, X_p\}$  be the set of  $p$  variables to be adjusted for. Individual  $i$  will have the observed covariate values denoted by the row vector  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})$ . An important assumption when modelling survival data is that of independent censoring, or non-informative censoring. This means that  $C$  is independent of  $T^*$  conditional on the variables  $X$ . That is, an individual whose survival time is censored at time  $c$  must be representative of all other individuals with those covariate values who have survived to that time [36]. Administrative censoring, as introduced in Section 1.1.1, can be assumed to be non-informative. Censoring during a study, for example, due to individuals being lost to follow-up, may not satisfy this assumption. Henceforth, let us assume censoring is independent.

The follow-up process for an individual may only begin at some known time after the time origin. This is called left truncation or delayed entry [36]. For example, if the time scale for an observational study is age, the participants will be left truncated at the age they enter the study, as this is when their follow-up process begins while their time origin is birth. Similar to the assumption of independent censoring, it is assumed that the left truncation time is independent of the event time. Let  $T_0$  be the random variable denoting the left truncation time, which takes the value of 0 if the follow-up process begins at the time origin. Let  $t_{0i}$  be the observed left truncation time for individual  $i$ .

Finally, consider  $n$  individuals. Let  $\mathbf{t} = (t_1, \dots, t_n)$ ,  $\boldsymbol{\delta} = (\delta_1, \dots, \delta_n)$  and  $\mathbf{t}_0 = (t_{01}, \dots, t_{0n})$  be the vector of the observed survival times, event indicator values and left truncation times for all  $n$  individuals, respectively. Let  $\mathbf{X}$  be the  $n \times p$  matrix of observed covariate values.

### 2.3.2 Likelihood Function

Models in which a specific probability distribution is assumed for the event times are known as parametric models [36]. The use of parametric models has been motivated in Section 1.1.1. This section will consider the following parametric models: exponential, Weibull, Gompertz, log-logistic, log-normal and Royston-Parmar models.

Once the distributional model for event times has been specified in terms of a probability density function,  $f(t)$ , the distributional parameters need to be estimated. This is often achieved by the method of maximum likelihood. The likelihood function is defined as the joint probability of the observed data, regarded as a function of the unknown parameters in the assumed model [36]. It is denoted as  $\mathcal{L}(\boldsymbol{\theta}|\mathbf{t}, \boldsymbol{\delta}, \mathbf{t}_0)$ , where  $\boldsymbol{\theta}$  is the vector of unknown parameters. The likelihood is the product of the likelihood contribution for each individual  $i$ ,  $L_i(\boldsymbol{\theta}|t_i, \delta_i, t_{0i})$ :

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{t}, \boldsymbol{\delta}, \mathbf{t}_0) = \prod_{i=1}^n L_i(\boldsymbol{\theta}|t_i, \delta_i, t_{0i})$$

First, let us consider the case of no left truncation. The contribution of individual  $i$  to the likelihood when there is no censoring (and therefore the survival time  $t_i$  is the event time  $t_i^*$ ) is:

$$L_i(\boldsymbol{\theta}|t_i^*) = f(t_i^*)$$

When censoring is present, the contribution of individual  $i$  to the likelihood is:

$$L_i(\boldsymbol{\theta}|t_i, \delta_i) = \{f(t_i)\}^{\delta_i} \{S(t_i)\}^{1-\delta_i}$$

The parameter estimates  $\hat{\boldsymbol{\theta}}$  that maximise the likelihood function are called the maximum likelihood estimates (MLEs). To calculate these, first the logarithm of

the likelihood function is taken. The log-likelihood function is the summation of the log-likelihood contribution for each individual  $i$ . The contribution of individual  $i$  to the log-likelihood,  $l_i(\boldsymbol{\theta}|t_i, \delta_i) = \log \{L_i(\boldsymbol{\theta}|t_i, \delta_i)\}$ , is:

$$l_i(\boldsymbol{\theta}|t_i, \delta_i) = \delta_i \{\log f(t_i)\} + (1 - \delta_i) \log \{S(t_i)\} \quad (2.6)$$

The log-likelihood function is then differentiated with respect to each parameter in  $\boldsymbol{\theta}$ . The values of  $\boldsymbol{\theta}$  that result in the derivatives simultaneously equating to zero are the MLEs. These can be obtained analytically where feasible (for example, the exponential model), or by using numerical routines.

Using Equations 2.3 and 2.4, the contribution of individual  $i$  to the log-likelihood can also be expressed as:

$$l_i(\boldsymbol{\theta}|t_i, \delta_i) = \delta_i \log \{h(t_i)\} - H(t_i) \quad (2.7)$$

Let us now also consider left truncation. The contribution of individual  $i$  to the log-likelihood in the presence of left truncation is:

$$l_i(\boldsymbol{\theta}|t_i, \delta_i, t_{0i}) = \delta_i \log \{h(t_i)\} - H(t_i) + H(t_{0i}) \quad (2.8)$$

For simplicity, left truncation is not included in the remainder of this chapter (except for multistate models); however, it is supported by all the methods discussed.

Covariates can be incorporated in parametric models, in which case the likelihood function will also depend on the observed covariate values  $\mathbf{X}$  and the regression coefficients will be included in  $\boldsymbol{\theta}$ . There are two common ways that covariates are included: by assuming proportional hazards or the AFT property. Section 2.3.3 describes parametric proportional hazards models, Section 2.3.4 considers an extension to these with Royston-Parmar models and Section 2.3.5 discusses AFT models. Examples of the log-likelihood function for different parametric models are also given.

### 2.3.3 Parametric Proportional Hazards Models

The first class of parametric models is proportional hazards models and these make an assumption on the relationship between the covariates and the hazard function. Let us first consider a single, binary variable  $X$ . Let  $h_1(t)$  be the hazard function when  $X = 1$  and let  $h_0(t)$  be the baseline hazard function, that is, when  $X = 0$ . Let  $\beta$  be the coefficient for the variable  $X$ . The proportional hazards assumption is such that the hazard function when  $X = 1$  is proportional to the hazard function when  $X = 0$ , for all points in time:

$$h_1(t) = h_0(t) \exp(\beta)$$

Under the proportional hazards assumption, the ratio of the hazard function for  $X = 1$  compared to  $X = 0$  (at all points in time) is the constant  $\exp(\beta)$ , denoted as the hazard ratio. The hazard ratio is a summary measure used to describe the increased or decreased rate of the event occurring associated with a change in variable  $X$ . This theory extends to categorical and continuous variables.

More generally, let  $X = \{X_1, \dots, X_p\}$  be the set of  $p$  variables to be adjusted for and let  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_p)$  be the row vector of  $p$  coefficients. Let  $h_0(t)$  be the baseline hazard function, that is, the hazard function when all variables are set to 0. Assuming proportional hazards, the hazard function at time  $t$  for the  $i^{th}$  individual is given by:

$$h_i(t) = h_0(t) \exp(\boldsymbol{\beta} \mathbf{x}_i^T) \quad (2.9)$$

Standard probability distributions that assume the proportional hazards property include the exponential, Weibull and Gompertz distributions. Estimates of  $\boldsymbol{\beta}$ , along with the corresponding distributional parameters, are found using maximum likelihood theory. This was introduced in Section 2.3.2. The log-likelihood is given for the aforementioned distributions, incorporating covariates, in the following paragraphs.

## Exponential Distribution

The exponential model is the simplest parametric survival model and assumes the hazard function is constant throughout time. It has one distributional parameter: the scale parameter  $\lambda > 0$ . The hazard function and cumulative hazard function are:

$$h(t) = \lambda$$

$$H(t) = \lambda t$$

The covariates are modelled by parameterising  $\lambda_i = \exp(\beta_0 + \boldsymbol{\beta} \mathbf{x}_i^T)$  for individual  $i$ .  $\exp(\beta_0)$  is the intercept term. Using Equation 2.7, the contribution to the log-likelihood for individual  $i$  is:

$$l_i(\boldsymbol{\beta}, \beta_0 | t_i, \delta_i, \mathbf{x}_i) = \delta_i \log(\lambda_i) - \lambda_i t_i \quad (2.10)$$

## Weibull Distribution

The Weibull model is the most widely used parametric model [36] and has a monotonically increasing or decreasing hazard function. It has two distributional parameters: the shape parameter  $\gamma > 0$  and scale parameter  $\lambda > 0$ . The survival function and probability density function are given here as well as the hazard function and cumulative hazard function, as the Weibull model is used frequently in the thesis:

$$S(t) = \exp(-\lambda t^\gamma) \quad (2.11)$$

$$f(t) = \lambda \gamma t^{\gamma-1} \exp(-\lambda t^\gamma)$$

$$h(t) = \lambda \gamma t^{\gamma-1}$$

$$H(t) = \lambda t^\gamma$$

The covariates are modelled by parameterising  $\lambda_i = \exp(\beta_0 + \boldsymbol{\beta} \mathbf{x}_i^T)$  for individual  $i$ . Using Equation 2.7, the contribution to the log-likelihood for individual  $i$

is:

$$l_i(\boldsymbol{\beta}, \beta_0, \gamma | t_i, \delta_i, \mathbf{x}_i) = \delta_i \log(\gamma \lambda_i t_i^{\gamma-1}) - \lambda_i t_i^\gamma \quad (2.12)$$

Note that when  $\gamma = 1$ , the Weibull model reduces to an exponential model. As  $\gamma$  is required to be positive, usually  $\log(\gamma)$  is estimated in the likelihood function. This is demonstrated in Section G.1.2.

### Gompertz Distribution

Like the Weibull model, the Gompertz model has a monotonically increasing or decreasing hazard function. It has two distributional parameters: the shape parameter  $\gamma$  and scale parameter  $\lambda > 0$ . The hazard function and cumulative hazard function are:

$$\begin{aligned} h(t) &= \lambda \exp(\gamma t) \\ H(t) &= \lambda \gamma^{-1} \{ \exp(\gamma t) - 1 \} \end{aligned}$$

The covariates are modelled by parameterising  $\lambda_i = \exp(\beta_0 + \boldsymbol{\beta} \mathbf{x}_i^T)$  for individual  $i$ . Using Equation 2.7, the contribution to the log-likelihood for individual  $i$  is:

$$l_i(\boldsymbol{\beta}, \beta_0, \gamma | t_i, \delta_i, \mathbf{x}_i) = \delta_i \{ \log(\lambda_i) + \gamma t_i \} - \lambda_i \gamma^{-1} \{ \exp(\gamma t_i) - 1 \} \quad (2.13)$$

Note that when  $\gamma = 0$ , the Gompertz model reduces to an exponential model [36].

It is important to check that the assumption of proportional hazards for each covariate is viable for the observed data. An informal method is to plot the log cumulative hazard function against log time for each level in a categorical covariate. Parallel lines indicate that the proportional hazards assumption is reasonable. Straight lines would suggest that a Weibull model is a reasonable distribution for the survival times. An alternative method to check the assumption is by including an interaction term between the covariate and time (or log time) in the model. If this term is statistically significant, it suggests that the hazards in the group are

not proportional throughout all time.

If the hazards are not proportional, one option is to incorporate the covariates with non-proportional hazards in the shape parameter  $\gamma$ . This can be done in a similar way to how the covariates were incorporated in the scale parameter  $\lambda$ . Alternatively, a more flexible parametric model could be used. The next section introduces Royston-Parmar models, which can address non-proportional hazards with time-dependent effects.

### 2.3.4 Royston-Parmar Models

One group of parametric models that are becoming increasingly used are Royston-Parmar models, see Royston and Parmar [8]. Royston-Parmar models can be a useful extension to standard probability distributions for survival data, as they provide additional flexibility that can accommodate complex shapes in the hazard function (with multiple turning points). They are also a useful alternative to proportional hazards models, when this assumption does not hold, as they can easily incorporate time-dependent effects. Royston-Parmar models will now be described following Lambert and Royston [42].

Royston-Parmar models utilise restricted cubic splines to model the log cumulative hazard function. A restricted cubic spline function of  $\log(t)$  with  $K$  knots,  $\mathbf{k}_0 = \{k_1, \dots, k_K\}$ , is used and involves the creation of  $K - 1$  derived variables,  $v_j \{\log(t)\}$ . For ease of exposition, the functional dependency of  $v_j \{\log(t)\}$  has been omitted from the notation. As in Section 2.3.3, let  $\boldsymbol{\beta}$  be the row vector of  $p$  coefficients for variables  $X$ . The log cumulative hazard function for a proportional hazards model is:

$$\begin{aligned} \log \{H(t|\mathbf{x}_i)\} &= \eta_i(t) = g\{\log(t)|\boldsymbol{\gamma}, \mathbf{k}_0\} + \boldsymbol{\beta}\mathbf{x}_i^T \\ &= \gamma_0 + \gamma_1 v_1 + \gamma_2 v_2 + \dots + \gamma_{K-1} v_{K-1} + \boldsymbol{\beta}\mathbf{x}_i^T \end{aligned}$$

Where the derived variables  $v_j$  (basis functions) are calculated as follows:

$$v_1 = \log(t)$$

$$v_j = \{\log(t) - k_j\}_+^3 - \phi_j \{\log(t) - k_1\}_+^3 - (1 - \phi_j) \{\log(t) - k_K\}_+^3 \quad j = 2, \dots, K - 1$$

$$\phi_j = (k_K - k_j)/(k_K - k_1)$$

And where  $x_+$  equals  $x$  if  $x > 0$  and equals 0 otherwise.

The survival and hazard functions are defined as:

$$S(t|\mathbf{x}_i) = \exp[-\exp\{\eta_i(t)\}]$$

$$h(t|\mathbf{x}_i) = \frac{dg\{\log(t)|\boldsymbol{\gamma}, \mathbf{k}_0\}}{dt} \exp\{\eta_i(t)\}$$

Where the derivatives of the restricted cubic spline functions are:

$$g'\{\log(t)|\boldsymbol{\gamma}, \mathbf{k}_0\} = g' = \gamma_1 v'_1 + \gamma_2 v'_2 + \dots + \gamma_{K-1} v'_{K-1}$$

$$v'_1 = \frac{1}{t}$$

$$v'_j = \frac{3}{t} [\{\log(t) - k_j\}_+^2 - \phi_j \{\log(t) - k_1\}_+^2 - (1 - \phi_j) \{\log(t) - k_K\}_+^2]$$

$$j = 2, \dots, K - 1$$

In the setting of time-independent effects,  $\boldsymbol{\beta}$  is a vector of log hazard ratios. Estimates of  $\boldsymbol{\beta}$  and  $\boldsymbol{\gamma}$  are found using maximum likelihood theory. Following Equation 2.7, the contribution to the log-likelihood for individual  $i$  is:

$$l_i(\boldsymbol{\beta}, \boldsymbol{\gamma}|t_i, \delta_i, \mathbf{x}_i, \mathbf{k}_0) = \delta_i (\log[g'\{\log(t_i)|\boldsymbol{\gamma}, \mathbf{k}_0\}] + \eta_i(t_i)) - \exp\{\eta_i(t_i)\} \quad (2.14)$$

As mentioned above, one of the main advantages of Royston-Parmar models is the ability to easily incorporate time-dependent effects. This is achieved by including interactions with the spline terms and covariates [42]. As before, let  $\mathbf{k}_0$  denote the knots associated with the baseline spline function. If there are  $D$  time-dependent effects, let  $\mathbf{k}_j$  and  $\boldsymbol{\delta}_j$  be the knots and coefficients associated with the  $j^{th}$  time-

dependent effect,  $j = 1, \dots, D$ , respectively. Let  $x_{ij}$  denote the observed value of the  $i^{th}$  individual for the  $j^{th}$  time-dependent effect. The log cumulative hazard function for a time-dependent effects Royston-Parmar model is then:

$$\log \{H(t|\mathbf{x}_i)\} = g\{\log(t)|\boldsymbol{\gamma}, \mathbf{k}_0\} + \prod_{j=1}^D g\{\log(t)|\boldsymbol{\delta}_j, \mathbf{k}_j\} x_{ij} + \boldsymbol{\beta} \mathbf{x}_i^T$$

A Royston-Parmar model with  $K$  knots will have 2 boundary knots,  $K - 2$  interior knots and  $K - 1$  degrees of freedom (DF). Note that a Royston-Parmar model with one degree of freedom is equivalent to the Weibull model. The number of knots  $K$  (and  $K_j$ ,  $j = 1, \dots, D$ , the number of knots for each time-dependent effect  $j$ ) and choice of the knot locations  $\mathbf{k}_0$  (and  $\mathbf{k}_j$ ,  $j = 1, \dots, D$ ) need to be chosen. The default interior knot locations suggested by Lambert and Royston [42], and programmed in `stpm2` [42], are at equally spaced centiles of the distribution of the uncensored log event times. The boundary knots are placed at the minimum and maximum of the uncensored log survival times.

Often, between three to five degrees of freedom are sufficient for the baseline spline function [43–45] and it has been demonstrated that RP models are not overly sensitive to this choice [43, 44]. Lambert and Royston [42] comment that, generally, the underlying shape of the baseline hazard is more complex than any departures from it and suggest fewer degrees of freedom for the time-dependent effects than the baseline spline function. Model fitting diagnostics, such as the Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), can be used to guide the choice of the degrees of freedom. As with all modelling, visual checks of the model fit are recommended and sensitivity analyses may be employed to confirm the estimates are robust to the choice of degrees of freedom.

### 2.3.5 Accelerated Failure Time Models

The second class of parametric models is AFT models. The covariates in AFT models are assumed to act multiplicatively on the time-scale and can be interpreted as affecting the speed of progression [36]. If the proportional hazards assumption

is not valid, these models might prove fruitful. Following Collett [36], let us first consider a single, binary variable  $X$ . Let  $S_1(t)$  be the survival function when  $X = 1$  and let  $S_0(t)$  be the baseline survival function, that is, when  $X = 0$ . Let  $\alpha$  be the coefficient for the variable  $X$ . The AFT assumption is such that the survival time when  $X = 1$  is a multiple of the survival time when  $X = 0$ , for all points in time:

$$S_1(t) = S_0 \left\{ \frac{t}{\exp(\alpha)} \right\}$$

Under the AFT assumption, the constant  $\exp(\alpha)$  is denoted as the time ratio. It can be interpreted as the survival time among individuals with  $X = 1$  is  $\exp(\alpha)$  times the survival time among individuals with  $X = 0$  [36]. If  $\exp(\alpha) < 1$ ,  $X = 1$  represents an accelerated time to the event relative to  $X = 0$ , while if  $\exp(\alpha) > 1$ ,  $X = 1$  represents an decelerated time to the event relative to  $X = 0$ . The quantity  $\exp(-\alpha)$  is therefore termed the acceleration factor [36]. This theory extends to categorical and continuous variables.

More generally, let  $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_p)$  be the row vector of  $p$  coefficients for the set of  $p$  variables  $X = \{X_1, \dots, X_p\}$ . Assuming the AFT assumption, the survival function at time  $t$  for the  $i^{th}$  individual is given by:

$$S_i(t) = S_0 \left\{ \frac{t}{\exp(\boldsymbol{\alpha} \mathbf{x}_i^T)} \right\}$$

Standard probability distributions that are used in AFT modelling include the log-logistic and log-normal distributions. These are named as such as the variable  $\log(T)$  has a logistic and normal distribution, respectively. When considering the log-linear model for the random variable  $T_i$ ,  $\log(T_i)$  is equal to the sum of an intercept, the coefficients  $\boldsymbol{\alpha}$  multiplied by the observed explanatory values  $\mathbf{x}_i^T$  and a scale parameter multiplied by the random variable  $\epsilon_i$ , which is used to model the deviation of the values of  $\log(T_i)$  from the linear part of the model.  $\epsilon_i$  is assumed to have a particular probability distribution and this determines the distribution of  $T_i$ . For a more detailed discussion see Sections 6.4-6.5 in Collett [36]. Note that the Weibull distribution can also be used in AFT modelling and, uniquely,

has both the proportional hazards and AFT properties [46]. Estimates of  $\boldsymbol{\alpha}$ , along with corresponding distributional parameters, are found using maximum likelihood theory. The log-likelihood is given for the log-logistic and log-normal distributions, incorporating covariates, in the following paragraphs.

### Log-logistic Distribution

Unlike the Weibull and Gompertz hazard functions, the log-logistic hazard function can accommodate a (single) change in direction. The distribution is called log-logistic because the variable  $\log(T)$  has a logistic distribution [36]. The log-logistic model has two distributional parameters:  $\gamma > 0$  and  $\lambda > 0$ . The survival function and density function are:

$$S(t) = \{1 + (\lambda t)^{1/\gamma}\}^{-1}$$

$$f(t) = \frac{\lambda^{1/\gamma} t^{1/\gamma-1}}{\gamma \{1 + (\lambda t)^{1/\gamma}\}^2}$$

The covariates are modelled by parameterising  $\lambda_i = \exp(-\alpha_0 - \boldsymbol{\alpha} \mathbf{x}_i^T)$  for individual  $i$ . Here,  $\exp(\alpha_0)$  is the intercept term. Using Equation 2.6, the contribution to the log-likelihood for individual  $i$  is:

$$l_i(\boldsymbol{\alpha}, \alpha_0, \gamma | t_i, \delta_i, \mathbf{x}_i) = \delta_i \left\{ \frac{1}{\gamma} \log(\lambda_i) + \left( \frac{1}{\gamma} - 1 \right) \log(t_i) - \log(\gamma) \right\} - \\ (\delta_i + 1) \log \left\{ 1 + (\lambda_i t_i)^{1/\gamma} \right\} \quad (2.15)$$

As  $\gamma$  is required to be positive, usually  $\log(\gamma)$  is estimated in the likelihood function. This is demonstrated in Section G.1.4.

### Log-normal Distribution

This distribution is called log-normal because the variable  $\log(T)$  has a normal distribution. The log-normal model has two distributional parameters: mean  $\mu$  and standard deviation  $\sigma > 0$ . The survival function and density function are the

following, where  $\Phi(\cdot)$  is the standard normal cumulative density function:

$$S(t) = 1 - \Phi \left\{ \frac{\log(t) - \mu}{\sigma} \right\}$$

$$f(t) = \frac{1}{t\sigma\sqrt{2\pi}} \exp \left[ \frac{-1}{2\sigma^2} \{ \log(t) - \mu \}^2 \right]$$

The covariates are modelled by parameterising  $\mu_i = \alpha_0 + \boldsymbol{\alpha}\mathbf{x}_i^T$  for individual  $i$ .

Using Equation 2.6, the contribution to the log-likelihood for individual  $i$  is:

$$l_i(\boldsymbol{\alpha}, \alpha_0, \sigma | t_i, \delta_i, \mathbf{x}_i) = \delta_i \left[ -\log(t_i) - \log(\sigma) - \frac{1}{2} \log(2\pi) - \frac{1}{2\sigma^2} \{ \log(t_i) - \mu_i \}^2 \right] + \\ (1 - \delta_i) \log \left[ 1 - \Phi \left\{ \frac{\log(t_i) - \mu_i}{\sigma} \right\} \right] \quad (2.16)$$

As  $\sigma$  is required to be positive, usually  $\log(\sigma)$  is estimated in the likelihood function. This is demonstrated in Section G.1.5.

In addition to the log-logistic and log-normal distributions, the gamma and generalised gamma distributions can be used in AFT modelling, see Collett [36] for details. More recently, flexible spline based AFT models have been proposed utilising restricted cubic splines [47] and regression B-splines [48].

As with proportional hazards models, the validity of the AFT assumption should be investigated. Graphically, a plot of the log-odds of survival beyond  $t$  against log time should result in a straight line if a log-logistic model is suitable [36]. Similarly, a plot of  $\Phi^{-1} \{ 1 - \widehat{S}(t) \}$  against log time should give a straight line if the log-normal model is suitable [36].

## 2.4 Other Analysis Approaches to Survival Data

### 2.4.1 Kaplan-Meier Estimator

The first step in many survival analyses is to calculate the Kaplan-Meier estimate of the survival function [49]. This may be the only analysis needed in the absence of covariates or may give an initial summary of the overall data if covariates are present.

It can be calculated for each level in a categorical variable separately. The Kaplan-Meier estimator can also be used to check graphically the fit of parametric models. This non-parametric estimator is a popular choice, as it makes no assumptions on the event times nor assumes a specific relationship between covariates and the event time [7]. The estimator is a step function where the survival estimate drops only when an individual experiences the event. Following Collett [36], the approach will now be outlined.

Suppose there are  $r$  event times among the  $n$  individuals, where  $r \leq n$ . Let the event times be arranged in ascending order and let the  $j^{th}$  event be denoted by  $t_{(j)}$  for  $j = 1, 2, \dots, r$ . Let  $n_j$  be the number of individuals who are event-free just before  $t_{(j)}$  and  $d_j$  be the number of events at  $t_{(j)}$ . In the case of a censored event time occurring at the same time as one or more events, the censored time is taken to occur immediately after the event time when computing the values of  $n_j$ . The Kaplan-Meier estimate of the survival function is then:

$$\hat{S}(t) = \prod_{j:t_{(j)} \leq t} \left( \frac{n_j - d_j}{n_j} \right) = \prod_{j:t_{(j)} \leq t} \left( 1 - \frac{d_j}{n_j} \right) \quad (2.17)$$

Where  $\hat{S}(t) = 1$  for  $t < t_{(1)}$ . If the largest observation is censored, let it be denoted by  $c^*$ , then  $\hat{S}(t)$  is undefined for  $t > c^*$ . Alternatively, if the largest observed time is uncensored,  $t_{(r)}$ ,  $\hat{S}(t)$  is zero for  $t \geq t_{(r)}$ .

The standard error (SE) of the Kaplan-Meier estimate of the survivor function is defined as follows and is known as Greenwood's formula [50]:

$$\text{se} \left\{ \hat{S}(t) \right\} \approx \hat{S}(t) \left\{ \sum_{j:t_{(j)} \leq t} \frac{d_j}{n_j(n_j - d_j)} \right\}^{\frac{1}{2}}$$

Confidence intervals can be constructed in the usual way using normal approximation. However, this may result in the upper/lower bound of the confidence interval being greater/lower than one/zero, respectively. As the survivor function is a probability, it (and its confidence interval) should be within the  $[0, 1]$  interval. This can be achieved by calculating the confidence interval on the  $\log(-\log(\cdot))$  scale

and then back-transforming to the original scale.

The primary limitation of the Kaplan-Meier estimator is that it can only stratify by categorical variables and, therefore, continuous variables cannot be adjusted for. If interest lies in a comparison between groups, it also cannot quantify the size of the covariate effect. However, hypothesis tests, such as the log-rank test, are available to test whether there is a difference in survival between groups, see Collett for details [36].

### 2.4.2 Nelson-Aalen Estimator

An alternative to the Kaplan-Meier estimator is the Nelson-Aalen estimator. The Nelson-Aalen estimator provides a non-parametric estimate of the cumulative hazard function. The corresponding estimate of the survival function can be obtained using Equation 2.4. The estimator was introduced by Nelson [51, 52] and extended by Aalen [53].

Following Collett [36] and the notation in Section 2.4.1, the Nelson-Aalen estimate of the cumulative hazard function is:

$$\tilde{H}(t) = \sum_{j:t_{(j)} \leq t} \frac{d_j}{n_j}$$

Note that tilde is used to emphasise that this is an alternate estimator to the Kaplan-Meier estimator. Using Equation 2.4, the Nelson-Aalen estimate of the survival function is:

$$\tilde{S}(t) = \prod_{j:t_{(j)} \leq t} \exp\left(-\frac{d_j}{n_j}\right)$$

The standard error of the Nelson-Aalen estimate for the survival function is:

$$\text{se}\{\tilde{S}(t)\} \approx \tilde{S}(t) \left( \prod_{j:t_{(j)} \leq t} \frac{d_j}{n_j^2} \right)^{\frac{1}{2}}$$

Alternatively, the variance can be calculated on the cumulative hazard scale and can be transformed to the survival scale.

The Nelson-Aalen estimator shares similar strengths and weaknesses with the Kaplan-Meier estimator, see Section 2.4.1. Furthermore, the Nelson-Aalen estimator has particular importance, as an extension of this estimator constitutes the non-parametric estimator in multistate survival models. This is discussed briefly in Section 2.7.3 and in detail in Section 7.4.

### 2.4.3 Cox Proportional Hazards Model

The Cox proportional hazards model [54] is a frequently used, semi-parametric approach that assumes proportional hazards. It is described here and included as a comparator in the thesis, due to its common usage. Like in Section 2.3.3, the hazard function at time  $t$  for the  $i^{th}$  individual is given by Equation 2.9. Unlike parametric models, no assumptions are made on the underlying shape of the baseline hazard function  $h_0(t)$ . To emphasise this, Equation 2.9 can be re-expressed as:

$$\log \left\{ \frac{h_i(t)}{h_0(t)} \right\} = \boldsymbol{\beta} \mathbf{x}_i^T$$

As before,  $\boldsymbol{\beta}$  is the vector of log hazard ratios, which can be interpreted as in Section 2.3.3, and is the primary metric of interest in a Cox model. The  $\boldsymbol{\beta}$  coefficients are found by maximising the partial likelihood function. The likelihood function for the Cox model is referred to as the partial likelihood because it depends only on the ranking of the event times rather than the (censored and uncensored) survival times themselves [36]. The process of obtaining the MLEs using the partial likelihood is described in the following paragraphs, following Collett [36].

Let the set of individuals who are at risk at time  $t_i$  (event-free and uncensored at a time just prior to  $t_i$ ) be denoted by  $R(t_i)$ . This is also known as the risk set. For now, let us assume that there are no ties; that is that only one individual has an event at each event time. Then, the partial likelihood is given as:

$$\mathcal{L}(\boldsymbol{\beta} | \mathbf{t}, \boldsymbol{\delta}, \mathbf{X}) = \prod_{i=1}^n \left\{ \frac{\exp(\boldsymbol{\beta} \mathbf{x}_i^T)}{\sum_{l \in R(t_i)} \exp(\boldsymbol{\beta} \mathbf{x}_l^T)} \right\}^{\delta_i}$$

The corresponding partial log-likelihood is:

$$\log \{ \mathcal{L}(\boldsymbol{\beta} | \mathbf{t}, \boldsymbol{\delta}, \mathbf{X}) \} = \sum_{i=1}^n \delta_i \left[ (\boldsymbol{\beta} \mathbf{x}_i^T) - \log \left\{ \sum_{l \in R(t_i)} \exp(\boldsymbol{\beta} \mathbf{x}_l^T) \right\} \right]$$

The maximisation of the partial likelihood generally requires numerical procedures. In the case of ties, the simplest approximation to the likelihood was given by Breslow [55] and an alternative was proposed by Efron [56], see references and Collett [36] for details.

The greatest advantage of the Cox model is that it can accommodate any shape in the baseline hazard function. However, the unspecified baseline hazard makes predictions of absolute risk (for example, survival probabilities) more difficult to calculate. If these quantities are to be estimated, an estimation of  $h_0(t)$  is required. Once  $\hat{\boldsymbol{\beta}}$  has been obtained,  $h_0(t)$  can be estimated using the method of Kalbfleisch and Prentice [57]. In the case of ties, the baseline hazard function can only be estimated using iterative methods [36]. This can be avoided by using an approximation such as the Breslow estimate of the baseline cumulative hazard function [36, 55]. These approaches are predominantly used to obtain an estimate of the survival function, which will be a step function. In large samples, the Breslow estimate will appear smooth (with lots of small steps) and the estimate of the survival function is a transformation of this. If interest lies in the baseline hazard function itself, a smoother can be used for the contributions to the hazard function, as this function is noisy and estimated at each event time. The hazard function will depend on the smoother chosen.

In the case of no covariates, the estimated survival function based on the baseline hazard estimate of Kalbfleisch and Prentice reduces to the Kaplan-Meier estimate [36]. Similarly, the estimated survival function based on the Breslow estimate of the baseline cumulative hazard function reduces to the Nelson-Aalen estimate of the survival function [36].

A number of residuals can be examined to investigate the fit of the Cox model and these include: Cox-Snell [58], martingale [59], deviance [60] and Schoenfeld

[61] residuals. As with parametric proportional hazards models, the proportional hazards assumption should be assessed for each covariate. Scaled Schoenfeld residuals, proposed by Grambsch and Therneau [62], are centred at  $\widehat{\beta}_l$  for each covariate  $l$ , where  $l = 1, \dots, p$ . If the proportional hazards assumption is appropriate, the gradient should be zero when these are plotted against functions of time [63]. See references for a discussion on model diagnostics for the Cox model [36, 64, 65].

## 2.5 Interval Censoring

### 2.5.1 Definitions and Notation

As introduced in Section 1.1.2, interval censoring arises when the event of interest occurs between two observed times, for example, between two assessment visits in a clinical trial. To some extent, many survival data are interval-censored, as some degree of rounding is often used, for example, to the nearest day or month. Let  $t_i^*$ , an observation of the random variable  $T^*$ , be the event time for individual  $i$ . Event time  $t_i^*$  is interval-censored if it is only known to occur in the interval  $[l_i, u_i]$ . This means that either  $l_i < t_i^* \leq u_i$ ,  $l_i \leq t_i^* < u_i$ ,  $l_i \leq t_i^* \leq u_i$  or  $l_i < t_i^* < u_i$ , depending on the context [3]. Many methods lead to the same results whether the interval is taken to be closed, open or half-open [3]. For the purpose of the simulation study in Section 3.5, the half-open interval  $(l_i, u_i]$  will be used, which represents that  $l_i < t_i^* \leq u_i$ .

When a dataset contains interval-censored data, it often also contains right-censored data. Let  $c_i$  be the (right) censoring time for individual  $i$ , as introduced in Section 2.3.1. In the context of a clinical trial, this would be the last assessment visit. If an event is right-censored, the event time  $t_i^*$  for individual  $i$  lies in the interval  $(c_i, \infty)$ . The dataset may also contain left-censored data, or data censored in the first interval. Let  $c_i^*$  be the left censoring time for individual  $i$ . In a clinical trial, this would be the first post-baseline assessment. If an event is left-censored, the event time  $t_i^*$  for individual  $i$  lies in the interval  $(0, c_i^*]$ . Figure 1.1 in Chapter 1 illustrates the censoring intervals.

The thesis will focus on singly interval-censored data. Doubly interval-censored data, which occurs when both the start and end time are censored, are outside the scope of the thesis. This section concentrates on the methods used to analysis interval-censored data. A brief review of the approaches is given and two methods, naive imputation and the likelihood-based approach, are described in more detail. Examples of interval-censored data in practice and a thorough exploration of naive imputation is given in Chapter 3.

### 2.5.2 Review of Analysis Methods

A commonly used method to address interval-censored data is to inappropriately use single, naive imputation techniques [3]. Once an event time has been imputed, the data is analysed as if it were observed exactly/right-censored. This can lead to biased results and incorrect inference [66, 67]. Naive imputation is discussed in the following subsection.

Despite the frequent use of naive imputation methods, many appropriate methods to address interval-censored data are available, including non-, semi- and fully parametric approaches. A brief review of some popular approaches are given here, while comprehensive coverage is given in the tutorials from Bogaerts *et al* [3], Lindsey and Ryan [67], Gómez *et al* [68] and Lesaffre [69].

Similar to the standard survival setting, maximum likelihood theory can be used to obtain parameter estimates for parametric survival models. The definition of the likelihood function is extended to include different types of censoring and details are given in Section 2.5.4. This method will be termed the (appropriate) likelihood-based approach in Chapter 3.

Another analysis approach for interval-censored data is the non-parametric maximum likelihood estimator, also known as the Turnbull estimator [70]. Unlike the Kaplan-Meier estimator, the non-parametric Turnbull estimate for the survival function for interval-censored data has no closed solution and must be calculated using an iterative formula [3]. First Peto [71], then Turnbull [70], noted that the maximum likelihood solution resulted in a set of intervals, where the estimated survival

function was constant outside of the intervals and unspecified within the intervals [3, 68]. More efficient algorithms than Turnbull’s expectation-maximisation (EM) algorithm, such as the iterative convex minorant (ICM) [72] and the EM-ICM [73], have been proposed as alternatives [68]; see references for details [3, 68].

In terms of semi-parametric methods, Finkelstein [74] first extended the Turnbull estimator to the proportional hazards model with interval censoring [3]. The method of Finkelstein was based on discrete survival times [3, 75]. This assumption is relaxed in the EM-type procedure proposed by Goetghebeur and Ryan [75], which reduces to a standard Cox proportional hazards model in the absence of interval censoring [69, 75]. Two alternative approaches are Farrington’s approach [76] and the Cox ICM algorithm [77]. Farrington’s approach [76] utilises generalised linear models, while Pan [77] reformulated the ICM algorithm so that covariates could be incorporated (which was not possible with the original ICM algorithm) [3].

The final approach discussed here is multiple imputation, which treats the interval-censored event times as a missing data problem. Similar to single imputation, a value is imputed and then more standard statistical analyses can be used on the right-censored data. Unlike naive imputation, this is done multiple times to take into account the uncertainty and Rubin’s rules [78] are followed to combine the estimators of the imputed datasets into one global estimator [3]. A number of multiple imputation algorithms have been proposed, primarily for analysing the Cox proportional hazards model for interval-censored data [79–81].

This review has focused on methods for interval-censored data in a frequentist framework; however, a range of Bayesian approaches are available, see Bogaerts *et al* [3] for details. The approaches reviewed here, along with others, feature in the literature review of simulation studies in Section 3.3.

### 2.5.3 Naive Imputation

As mentioned in Section 1.1.2, there are three naive imputation methods that can be used to simplify interval-censored data to exactly observed/right-censored data: beginning (left), midpoint and end (right) imputation. In all cases, right-censored

data remains right-censored. As in Section 2.5.1, let the event time  $t_i^*$  for individual  $i$  lie in the interval  $(l_i, u_i]$ . Each imputation approach is now described:

- **Beginning Imputation:** is when the event time is imputed as  $l_i$  for individual  $i$ , which is the beginning of the interval. Consideration is required if the event occurs in the first interval (left-censored) as  $l_i = 0$  and event times are required to be positive. One solution is to use a small increment; however, this raises the question of how small. This imputation method is rarely used in practice.
- **Midpoint Imputation:** is when the event time is imputed as  $\frac{l_i+u_i}{2}$  for individual  $i$ , which is the midpoint of the interval. This approach is used in practice and is more common in observational studies.
- **End Imputation:** is when the event time is imputed as  $u_i$  for individual  $i$ , which is the end of the interval. This approach is used frequently in clinical trials, especially in oncology.

#### 2.5.4 Maximum Likelihood Estimation

As with exactly observed/right-censored data in Section 2.3.2, maximum likelihood theory can be used to appropriately obtain parameter estimates for interval-censored data. The contribution to the likelihood function depends on the type of censoring and is shown in Table 2.1.

**Table 2.1:** Contribution from individual  $i$  to the likelihood function for different types of censoring (an extension of Table 1.1 from [3] using Equation 1.13 from [3])

Observation	Limits of the interval	Contribution to the likelihood
Left-censored in time $u_i$	$0 = l_i < u_i < \infty$	$1 - S_i(u_i)$
Right-censored in time $l_i$	$0 < l_i < u_i = \infty$	$S_i(l_i)$
Interval-censored in time $[l_i, u_i]$	$0 < l_i < u_i < \infty$	$S_i(l_i) - S_i(u_i)$
Exactly observed time $t_i$	$0 < l_i = u_i = t_i^* < \infty$	$f_i(t_i)$

First, consider the case where data are left-, interval- or right-censored. Let there be  $l$  and  $r$  left- and right-censored events, out of the total  $n$  events, respectively. Let

the individuals  $i$  be ordered such that all individuals who are left-censored are first, followed by right-censored individuals and, lastly, interval-censored individuals. Let  $\mathbf{l} = (l_1, \dots, l_n)$  be the vector of interval start times, where this value is missing (or 0) for individuals with left-censored data. Let  $\mathbf{u} = (u_1, \dots, u_n)$  be the vector of interval end times, where this value is missing (or infinity) for individuals with right-censored data. The overall likelihood function can be written as [3, 36]:

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{l}, \mathbf{u}) = \prod_{i=1}^l \{1 - S(u_i)\} \prod_{i=l+1}^{l+r} S(l_i) \prod_{i=l+r+1}^n \{S(l_i) - S(u_i)\}$$

Now, consider the case where data can also be observed exactly. Let there be  $e$  exactly observed events. Let the individuals  $i$  be ordered such that all individuals who are left-censored are first, followed by right-censored individuals, followed by individuals with exactly observed events and, lastly, interval-censored individuals. Note that if the event is observed exactly for individual  $i$ , then  $t_i^* = l_i = u_i$ . The overall likelihood function can be written as:

$$\mathcal{L}(\boldsymbol{\theta}|\mathbf{l}, \mathbf{u}) = \prod_{i=1}^l \{1 - S(u_i)\} \prod_{i=l+1}^{l+r} S(l_i) \prod_{i=l+r+1}^{l+r+e} f(t_i^*) \prod_{i=l+r+e+1}^n \{S(l_i) - S(u_i)\}$$

Parameters estimates  $\hat{\boldsymbol{\theta}}$  that maximise the likelihood function are then found, similar to when data are exactly observed/right-censored. A number of parametric models can be used and covariates incorporated, as before.

## 2.6 Inverse Probability Weighting

### 2.6.1 Definitions and Notation

Following the framework and notation of Williamson *et al* [82], let  $Z$  denote the binary treatment allocation/exposure variable ( $0 =$  control/no exposure,  $1 =$  treatment/exposure). For this section, let  $X = \{X_1, \dots, X_p\}$  be the set of  $p$  measured (baseline) confounders to be adjusted for. As defined in Section 2.3.1, the variable  $T = \min(T^*, C)$  and  $\delta$  is the event indicator.

Let  $z_i$  be the observed treatment value for individual  $i$  and  $t_i$  be the observed survival time. Let  $\delta_i$  be the observed event indicator for individual  $i$ . Let  $\mathbf{X}$  be the  $n \times (p + 1)$  matrix of confounder values with an additional column of ones for the intercept term and let  $\mathbf{x}_i = (x_{i0}, x_{i1}, \dots, x_{ip})$  be the vector of observed confounder values (with  $x_{i0} = 1$  for the intercept) for individual  $i$ .

Let  $T^1$  be the survival time that would have been observed under treatment ( $Z = 1$ ) and  $T^0$  be the survival time that would have been observed under no treatment ( $Z = 0$ ). The variables  $T^1$  and  $T^0$  are called counterfactual (or potential) outcomes as they represent outcomes that may not occur [15]. In causal inference, interest lies in contrasts between the counterfactual outcomes, also called contrasts in marginal estimands or population causal effects [15].

### 2.6.2 Contrasts in Marginal Estimands

In survival analysis, interest may lie in the following contrasts in marginal estimands, where  $t$  is the time at which the prediction is made [20]:

1. Difference in marginal survival probability at time  $t$ :

$$\begin{aligned}\Delta_S(t) &= E\{1(T^1 > t)\} - E\{1(T^0 > t)\} \\ &= P(T^1 > t) - P(T^0 > t)\end{aligned}\tag{2.18}$$

2. Difference in marginal RMST at time  $t$  [83]:

$$\begin{aligned}\Delta_\mu(t) &= E\{\min(T^1, t)\} - E\{\min(T^0, t)\} \\ &= \int_0^t \{P(T^1 > u) - P(T^0 > u)\} du\end{aligned}\tag{2.19}$$

3. Ratio of the marginal hazard functions at time  $t$ : Where  $h^z$  is the marginal hazard function corresponding to the counterfactual outcome under

treatment level  $Z = z$ .

$$\begin{aligned}\Delta_{HR}(t) &= \frac{h^1(t)}{h^0(t)} & (2.20) \\ h^z(t) &= \frac{d}{dt} (-\log [\mathbb{E} \{1(T^z > t)\}]) \quad z = 0, 1 \\ &= \frac{d}{dt} [-\log \{\mathbb{P}(T^z > t)\}] \quad z = 0, 1\end{aligned}$$

The marginal hazard functions can be assumed to be proportional (at all points in time), resulting in a marginal hazard ratio  $\Delta_{HR}(t) = e^\beta$ .

Mao *et al* [20] gives an overview of contrasts in marginal estimands in a survival setting. Differences in marginal RMST and survival probabilities are the preferred population causal effects, as the marginal hazard ratio, although frequently reported, suffers from two main issues. The first is that, generally, the hazard ratio changes over time. However, often studies report a single averaged estimate over the duration of follow-up [15, 84]. This means that the result may differ depending on what follow-up time the hazard ratio is estimated at [84] and that the hazard ratio may be 1 even if the survival curves are not identical [15].

Time-specific hazard ratios may alleviate the first problem; however, the hazard ratio still suffers from the second issue of selection bias [15, 84]. The hazard at time 2, for example, is the probability of experiencing the event at time 2, conditioning on those alive at time 1. In the presence of an unmeasured variable that affects survival (at all time points), but not treatment, conditioning on being alive at time 1 opens a backdoor pathway between the treatment and survival at time 2 [15]. This can induce an association between treatment and survival at time 2 [15]. Hernan and Robins [15], Technical Point 8.1 and Fine Point 17.2, discuss this issue in more detail and Hernan [84] gives an example. The hazard ratio is also not collapsible [85]. That is, in the absence of confounding, the conditional and marginal hazard ratios do not coincide [86].

The above contrasts in marginal estimands require knowledge of both counterfactual outcomes; however, only one will be observed for each individual. These contrasts in marginal quantities can be estimated using IP weighting under assump-

tions.

### 2.6.3 Assumptions

Three assumptions, collectively known as the identifiability conditions, are required for causal inference using IP weighting: consistency, conditional exchangeability and positivity. When estimates are obtained via models, the additional assumption of no model misspecification is required. See Hernan and Robins [15], Chapter 3, and Cole and Hernan [87] for a discussion on the main assumptions when performing an IP weighted analysis. The four main assumptions are discussed below.

In addition, the standard assumption of independent censoring conditional on treatment/exposure variable  $Z$  is made, along with the assumption of no interference, which is included in the Stable Unit Treatment Value Assumption (SUTVA) described by Rubin [88]. That is, that an individual's counterfactual outcome under treatment does not depend on other individual's treatment values [15].

#### Consistency

The consistency assumption requires that if an individual received treatment  $Z = z$ , then their observed outcome equals the counterfactual outcome under treatment  $z$ , that is  $T = T^z$  [15]. Hernan and Robins [15] break this into two components: precisely defining counterfactuals  $T^z$  by treatment  $Z$  being well defined and by linking the counterfactuals to the observed outcomes. The first component is necessary to avoid situations where the value of counterfactual  $T^z$  may differ depending on a version of treatment  $Z = z$  they receive [15]. For example, different surgeons administering treatment  $Z = z$  may result in a different value of counterfactual outcome  $T^z$ . The assumption of no multiple versions of treatment alleviates this issue [15].

The second component involves ensuring that only individuals receiving treatment  $Z = 1$  are considered as treated individuals in the population, and similarly for untreated [15]. The authors [15] stress the importance of defining treatments with sufficient detail (so that the causal effect estimates can be easily interpreted)

without excess constraints (so that there are enough examples of the treatment in the data). This can be facilitated by assuming treatment variation irrelevance [89], instead of no multiple versions of treatments [15]. The assumption requires the counterfactual outcome  $T^z$  to have the same value for treatment  $Z = z$ , regardless of which version of treatment  $z$  is received.

### Conditional Exchangeability

The conditional exchangeability assumption requires that the conditional probability of receiving treatment  $Z = z$  (for all possible values of  $z$ ) depends only on the measured confounders  $X$  [15]. This is often referred to as the no unmeasured confounding assumption. More formally, it is assumed that  $T^z \perp\!\!\!\perp Z|X, \forall z$ , that is, the counterfactual outcomes  $T^z$  are independent of the treatment  $Z$  conditional on the measured confounders  $X$  [15].

As this assumption is not testable, analysts are required to rely on expert knowledge to ensure all confounders are measured. It may be tempting to include many potential confounders; however, this may induce or amplify bias, see Section 2.6.5. Cole and Hernan [87] also warn that increasing the number of potential confounders may affect the validity of the positivity assumption. Adding a non-confounder may also reduce the statistical efficiency and result in estimates with wider confidence intervals [90].

### Positivity

The positivity assumption requires there being a non-zero (that is, positive) probability of receiving each level of treatment in every observed subset of confounders  $X$  [15]. More formally  $P(Z = z|X = x) > 0 \forall z$  and  $\forall x$  where  $P(X = x) \neq 0$  [15]. Positivity is only required for the confounders  $X$  that are required for conditional exchangeability [15].

Violations of this assumption can arise by structural zeros and random zeros [87]. The first is when an individual cannot possibly receive treatment  $Z = z$  at one or more observed subsets of the confounders. The second may occur by chance in small

sample sizes and/or when  $X$  is of high dimension (for example includes continuous variables). Parametric models can smooth over the random zeros. This assumption can be informally tested by checking the estimated probability of receiving each treatment for all subsets of observed confounder values  $X = x$  is greater than 0 and less than 1.

### No Model Misspecification

When models are used to perform an analysis (rather than using empirical approaches), it is required that there is no model misspecification for valid inference. This relates to both the treatment model and outcome model. For the treatment model, this means that all necessary interaction terms are included and the functional form of covariate effects are correct. The outcome model is also required to be correct. For survival models, this means that the baseline hazard has to be correctly defined (for parametric models) and any assumptions on the treatment effect have to be valid (for example proportional hazards).

An extension to IP weighting that allows more leniency in model misspecification is doubly robust methods. Doubly robust methods are a hybrid of IP weighting and regression standardisation (or g-computation), where both the treatment and outcome are modelled against the confounders [15]. Only one of the model specifications is required to be correct to give a consistent estimate of the causal effect [15]. There are different types of doubly robust estimators, for an example see Bang and Robins [91].

#### 2.6.4 Inverse Probability Weighted Analysis Algorithm

As discussed in Section 1.1.3, the purpose of IP weighting is to create a pseudo-population where the distribution of the measured baseline confounders is independent of the treatment assignment [14]. If the assumptions described in Section 2.6.3 hold, it allows the analyst to estimate the contrasts in marginal estimands introduced in Section 2.6.2.

There are three main components when performing an IP weighted analysis on survival data, which are discussed in the following three sections:

1. **Treatment model:** The treatment variable  $Z$  is modelled incorporating the measured confounders  $X$ , in order to estimate the propensity score. This is usually done with logistic regression and is discussed in Section 2.6.5.
2. **Weights:** The weights are calculated using the estimated propensity score. Two weights are commonly used for targeting causal effects in the whole population: stabilised and unstabilised weights. Weighting strategies are discussed in Section 2.6.6.
3. **Outcome model:** A weighted survival model is used to model survival outcome  $T$  with treatment  $Z$  as the only covariate and this is discussed in Section 2.6.7.

Point estimation for IP weighted survival models is explored in Chapter 4, while variance estimation is considered in Chapter 5.

### 2.6.5 Treatment Model: Modelling the Propensity Score

The first step to modelling the treatment variable  $Z$  is to determine what confounders  $X$  should be adjusted for.  $X$  is a confounder if it is associated with both the treatment and outcome. More formally,  $X$  is a confounder if there is conditional exchangeability  $T^z \perp\!\!\!\perp Z|X$  but not unconditional exchangeability  $T^z \perp\!\!\!\perp Z$  [15]. Drawing a directed acyclic graph may help to identify confounders, Chapter 6 [15], along with utilising expert knowledge and reviewing the existing literature. Adjusting for variables  $M$  that are not confounders can induce or amplify bias, Chapter 18 [15], for example, when:

- $M$  is a **collider**. Variable  $M$  is a collider if it is a common effect of  $T$  and  $Z$  [15]. A general example is if haplotype  $Z$  has no causal effect on someone's risk of being a cigarette smoker  $Y$  (here  $Y$  represents a binary outcome), but both  $Z$  and  $Y$  have a causal effect on the risk of heart disease  $M$  [15].

Conditioning on a collider can open a backdoor pathway between  $Z$  and  $T$ , inducing selection bias [15].

- $M$  is a **mediator**. Variable  $M$  is a mediator if it is affected by  $Z$  and affects  $T$  [15]. A general example is if aspirin  $Z$  affects the risk of heart disease  $Y$  (here  $Y$  represents a binary outcome) through its effect on platelet aggregation  $M$  [15]. Conditioning on a mediator blocks the component of the effect that goes through  $M$  and results in over-adjustment [15].
- $M$  is an **instrument**. Variable  $M$  is an instrument if it is associated with  $Z$ , does not affect  $T$  except through its potential effect on  $Z$  and if  $M$  and  $T$  do not share causes [15]. An example is in randomised clinical trials where  $M$  is the randomised treatment assignment,  $Z$  is the treatment actually received and  $T$  is the outcome [15]. Adjusting for an instrument can amplify the bias that arises from unmeasured confounders [15].

As mentioned in Section 2.6.3, including non-confounders may also reduce the statistical efficiency (and result in wider confidence intervals) [90] and including too many potential confounders may endanger the positivity assumption. If faced with many potential confounders, model selection processes, such as forward selection, backward elimination and stepwise selection, may not necessarily be relevant as these are designed for prediction rather than causal inference [15]. One suggestion has been to use machine learning techniques, such as the lasso, tree-based algorithm and neural networks, Chapter 18 [15]. However, the predictive machine learning algorithms do not guarantee the selected variables will eliminate confounding [15].

Let us assume that the confounders  $X$  have been measured and selected. In order to estimate the weights, the propensity score needs to be estimated. The propensity score is the probability an individual receives the treatment/exposure ( $Z = 1$ ), conditional on their observed covariate values  $\mathbf{x}_i$  and is denoted for individual  $i$  by  $e_i = e(\mathbf{x}_i) = P(Z = 1|\mathbf{x}_i)$ . The binary treatment/exposure variable  $Z$  can be modelled using logistic regression with parameters  $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \dots, \alpha_p)$ . The treatment model needs to be correctly specified with consideration to possible

interaction terms and the functional forms of the covariate effects. For simplicity, a main effects model is shown here. The propensity score for individual  $i$  can then be estimated using the MLEs  $\hat{\alpha}$ :

$$\hat{e}_i = \hat{e}(\mathbf{x}_i) = \frac{\exp(\hat{\alpha}\mathbf{x}_i^T)}{1 + \exp(\hat{\alpha}\mathbf{x}_i^T)} \quad (2.21)$$

A logistic regression model is used to model treatment in the thesis; however, it is possible to use other methods to estimate the propensity score, such as the probit model or machine learning techniques (as mentioned above). The thesis focuses on a binary treatment variable; however, a multinomial or nested logistic regression model can be used for a categorical treatment variable and ordinal logistic regression can be used for an ordinal treatment [92, 93]. For continuous treatments, one needs to model a probability density function, rather than a probability, increasing the difficulty and the sensitivity of results to the assumed probability density function, Chapter 12 [15].

For stabilised weights, the (unconditional) probability of treatment  $\pi = P(Z = 1)$  is required. This is estimated using  $\hat{\alpha}_S$ , which is the MLE of the intercept in a logistic model for treatment/exposure with no covariates:

$$\hat{\pi} = \frac{\exp(\hat{\alpha}_S)}{1 + \exp(\hat{\alpha}_S)} \quad (2.22)$$

### 2.6.6 Weights

As discussed in Section 1.1.3, when the target population is the whole population, two weights are commonly used: unstabilised and stabilised weights. Unstabilised weights are the inverse of the conditional probability of receiving the treatment level that was actually received. Unstabilised weights, denoted by  $u_i$  for individual  $i$ , can be estimated from the estimated propensity score by:

$$\hat{u}_i = \frac{z_i}{\hat{e}_i} + \frac{1 - z_i}{1 - \hat{e}_i} \quad (2.23)$$

It is useful to check the distribution of the estimated propensity scores and weights, for example, propensity scores near 0 or 1 may suggest non-positivity. The average of the unstabilised weights should be 2, as the pseudo-population should be double the size of the original with each individual having both counterfactuals represented (one observed and one estimated). Austin and Stuart [94] have suggested a set of balancing diagnostics to check the measured confounders are balanced between the two treatment groups in the weighted population. The diagnostics include the weighted standardised difference to compare means, prevalences, higher-order moments and interactions, as well as graphical methods to compare the distribution of continuous variables between the treatment groups.

Stabilised weights can be used instead of unstabilised weights to protect against increased variability that may arise from extreme weights. Stabilised weights multiply the unstabilised weights by the unconditional probability of receiving the treatment level that was actually received. Stabilised weights, denoted by  $s_i$  for individual  $i$ , can be estimated from the estimated propensity score by:

$$\hat{s}_i = \frac{\hat{\pi}z_i}{\hat{e}_i} + \frac{(1 - \hat{\pi})(1 - z_i)}{1 - \hat{e}_i} \quad (2.24)$$

The mean of the stabilised weights should be 1, as the pseudo-population should be scaled down to be the same size as the original data. An average weight far from 1 may indicate model misspecification or violations of positivity [15, 87].

An additional method that has been considered to address extreme weights is truncated, or trimmed, weights [87]. This is performed by replacing weights above (below) percentile  $p$  ( $100 - p$ ) to the value of the  $p$  ( $100 - p$ ) percentile [87], respectively, for example the  $1^{st}$  and  $99^{th}$  percentiles when  $p = 99$ . Cole and Hernan [87] explored the bias-variance trade-off, where increasing the amount of truncation (increasing  $p$ ) led to more precise but more biased estimates in their example. Lee *et al* [95] also investigated truncated weights. They concluded that more attention should be given to improving model specification in the case of extreme weights, rather than relying on weight truncation. Truncating weights will also mean that

the whole sampled population is no longer being targeted.

Other weighting strategies can be used to target other populations that may be of interest. Briefly, these include the average treatment effect in the treated weight [16], the average treatment effect in the controls weight [20], the matching weight [96] (which leads to an estimand that is analogous to the one obtained from one-to-one caliper matching without replacement [20]) and the overlap weight [97] (which can lead to an exact balance of the covariates [20]). See Mao *et al* [20] for an exploration of the general class of balancing weights.

### 2.6.7 Outcome Model: Inverse Probability Weighted Survival Model

Once the weights have been estimated, fully, semi- and non-parametric methods can be used to analyse the IP weighted survival data. This involves fitting a marginal model to the weighted survival data where treatment is the only covariate. The term ‘marginal’ often refers to a method where the estimates are unconditional on covariates [98]. Although this is often true in the case of IP weighting, the original intended meaning of ‘marginal’ in this context was to signify that the marginal distribution of the potential outcomes are being modelled [18, 98]. For more information see Breskin *et al* [98].

The primary focus of the thesis is on IP weighted parametric survival models, although non- and semi-parametric approaches are mentioned in Chapters 4 and 5. Parametric models are considered in detail as they more naturally extend to a range of contrasts in marginal estimands, for example difference in marginal RMST and survival probabilities, than the Cox model. It may be desired to use these contrasts in marginal estimands given the potential issues with the (marginal) hazard ratio, see Section 2.6.2. However, as mentioned in Section 2.6.3, valid inference relies on no model misspecification, which applies to the choice of outcome survival model.

An IP weighted parametric model with parameters  $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_q)$  is now considered. At least one coefficient in  $\boldsymbol{\beta}$  corresponds to the treatment effect (for

example,  $\beta_1$ ), although there may be more than one if treatment is modelled as an ancillary parameter or with a time-dependent effect (for example, in Royston-Parmar models). The MLEs  $\hat{\beta}$  can be obtained by maximising the weighted log-likelihood (weighted with unstabilised or stabilised weights). Let  $l_i^w$  denote the contribution of individual  $i$  to the weighted log-likelihood of the outcome survival model with weight  $w_i$ . Weight  $w_i$  is either stabilised, Equation 2.24, or unstabilised, Equation 2.23. The contribution of individual  $i$  to the weighted log-likelihood is then:

$$l_i^w(\beta|t_i, \delta_i, z_i, w_i) = w_i [\delta_i \log \{h(t_i)\} - H(t_i)] \quad w_i = s_i, u_i \quad (2.25)$$

Contrasts in the marginal estimands, such as the difference in marginal survival (Equation 2.18) or RMST (Equation 2.19), can then be estimated by the appropriate function of the parameter estimates from the weighted survival model. This can be done analytically (where feasible) or by using numerical integration.

## 2.7 Multistate Survival Models

### 2.7.1 Definitions, Estimands and the Markov Assumption

Following Fiocco *et al* [99] and Crowther and Lambert [22], consider a stochastic process  $Y(t)$ ,  $t \geq 0$ , with a finite state space  $\mathcal{Z} = \{1, \dots, Z\}$  and process history up to time  $s$ ,  $\mathcal{H}_s = \{Y(u); 0 \leq u \leq s\}$ . The transition probabilities can then be defined as:

$$\text{P}\{Y(t) = b | Y(s) = a, \mathcal{H}_{s-}\} \quad (2.26)$$

The transition probability is the probability that an individual in state  $a$  at time  $s$  moves to state  $b$  by time  $t$ , conditional on the process history up until the time just before  $s$ . This can be simplified to a Markov model (Markov process), which makes the assumption that the probability in Equation 2.26 is only conditional on the state at time  $s$  and no other process history:

$$P_{ab}(s, t) = \text{P}\{Y(t) = b | Y(s) = a, \mathcal{H}_{s-}\} = \text{P}\{Y(t) = b | Y(s) = a\} \quad (2.27)$$

Assuming a Markov model is a commonly used simplification as transition probabilities can then be obtained by solving the forward Kolmogorov equations (either analytically in the case of the exponential model or numerically for more complex distributions). This assumption can be relaxed to give a semi-Markov model, where the transition probabilities can depend on the time at which earlier states were entered [22]. The remainder of the thesis focuses on Markov models.

Another important metric is the transition rate, or cause-specific hazard rate, from state  $a$  to state  $b$  at time  $t$  and is defined as:

$$h_{ab}(t) = \lim_{\delta t \rightarrow 0} \left[ \frac{P \{Y(t + \delta t) = b | Y(t) = a\}}{\delta t} \right] \quad (2.28)$$

This represents the instantaneous rate of moving from state  $a$  to  $b$  and is analogous to the hazard rate in the standard survival setting (Equation 2.2). The transition rates can also be indexed by  $k$  where the  $k^{th}$  transition goes from state  $a_k$  to state  $b_k$ :

$$h_k(t) = \lim_{\delta t \rightarrow 0} \left[ \frac{P \{Y(t + \delta t) = b_k | Y(t) = a_k\}}{\delta t} \right] \quad (2.29)$$

The collection of transition rates governs the rate at which individuals move between states and therefore the multi-state model. For a tutorial in multi-state models see Putter *et al* [1]. Note that multi-state models can be uni-directional (states can never be returned to once left) or bi-directional (it is possible to return to at least one state once it has been left).

Another useful measure is the expected length of stay in a state. This is analogous to MST in the standard survival setting (Equation 2.5). A state  $a$  is defined as an absorbing state if once an individual enters state  $a$ , they will never leave it. More formally,  $a$  is an absorbing state if  $P \{Y(t) = b | Y(s) = a\} = 0 \ \forall b \neq a \ \forall t > s$ . The expected length of stay in state  $b$  during the interval  $[0, \infty)$ , given an individual starts in non-absorbing state  $a$  is [100]:

$$e_{ab} = \int_0^\infty P \{Y(u) = b | Y(0) = a\} du \quad (2.30)$$

If the integral in Equation 2.30 is bounded up to time  $t$  (instead of  $\infty$ ), this is known as restricted length of stay. Often the last event time is chosen. Any time chosen after the last observed time can lead to issues with extrapolation. If the integral in Equation 2.30 begins from time  $s$  (instead of 0), then this is known as residual length of stay. The restricted, residual length of stay in state  $b$  up to time  $t$ , given an individual is in non-absorbing state  $a$  at time  $s$  is [100]:

$$e_{ab}(s, t) = \int_s^t P\{Y(u) = b | Y(s) = a\} du \quad (2.31)$$

See Grand and Putter [100] for more details on expected length of stay.

### 2.7.2 Modelling the Transitions

Let there be  $K$  transitions in the multistate model. Let there be  $n_k$  individuals,  $j = 1, \dots, n_k$ , who are at risk of the  $k^{th}$  transition. Let  $\mathbf{t}_k$ ,  $\mathbf{t}_{0k}$  and  $\boldsymbol{\delta}_k$ ,  $k = 1, \dots, K$ , be vectors of length  $n_k$ .  $\mathbf{t}_k$ ,  $\mathbf{t}_{0k}$  and  $\boldsymbol{\delta}_k$  denote the survival times, left truncation times and transition indicators for the  $k^{th}$  transition, respectively. Let  $t_{jk}$  be the survival time for the  $j^{th}$  individual for the  $k^{th}$  transition, where  $t_{jk}$  is the  $j^{th}$  element of vector  $\mathbf{t}_k$ . In other words, this is the time individual  $j$  left state  $a_k$  or was censored. They may go to state  $b_k$  (experienced transition  $k$ ,  $\delta_{jk} = 1$ ) or go to another state/be right-censored (censored for transition  $k$ ,  $\delta_{jk} = 0$ ).  $\delta_{jk}$  therefore indicates whether individual  $j$  experienced transition  $k$  and is the  $j^{th}$  element of  $\boldsymbol{\delta}_k$ . Finally,  $t_{0jk}$  is the time the  $j^{th}$  individual enters state  $a_k$  and is the  $j^{th}$  element of  $\mathbf{t}_{0k}$ .

Parameter estimates from parametric multistate models can be obtained by maximising the likelihood function. As before, let  $\boldsymbol{\theta}$  be the vector of parameters to be estimated. The likelihood function for a multistate model can be written as a product over the  $K$  transitions and the  $n_k$  individuals at risk of transition  $k$  [101]. The likelihood function is then:

$$\mathcal{L}(\boldsymbol{\theta} | \mathbf{t}_1, \dots, \mathbf{t}_K, \boldsymbol{\delta}_1, \dots, \boldsymbol{\delta}_K, \mathbf{t}_{01}, \dots, \mathbf{t}_{0K}) = \prod_{k=1}^K \prod_{j=1}^{n_k} L_{jk}(\boldsymbol{\theta} | t_{jk}, \delta_{jk}, t_{0jk}) \quad (2.32)$$

For ease, let us consider the contribution of individual  $j$  to the log-likelihood for the  $k^{th}$  transition:  $l_{jk}(\boldsymbol{\theta}|t_{jk}, \delta_{jk}, t_{0jk}) = \log \{L_{jk}(\boldsymbol{\theta}|t_{jk}, \delta_{jk}, t_{0jk})\}$ . The log-likelihood function is a summation of the individual contributions over the transitions and individuals at risk of each transition. Equation 2.8 in the standard survival setting can then be extended to take into account the  $k^{th}$  transition, where  $h_k(t)$  and  $H_k(t)$  are the hazard and cumulative hazard function for the  $k^{th}$  transition, respectively:

$$l_{jk}(\boldsymbol{\theta}|t_{jk}, \delta_{jk}, t_{0jk}) = \delta_{jk} \log \{h_k(t_{jk})\} - H_k(t_{jk}) + H_k(t_{0jk})$$

As in Section 2.3.2, if the parameter vector  $\boldsymbol{\theta}$  includes covariate coefficients, the (log-)likelihood function also depends on the observed covariate values  $\mathbf{X}$ .

If the parameter vector  $\boldsymbol{\theta}$  can be partitioned as  $(\boldsymbol{\theta}_1|...|\boldsymbol{\theta}_K)$ , that is,  $K$  independent sets of parameters with each one corresponding to each transition  $k$ , the likelihood becomes the product of  $K$  independent transition-specific likelihoods [101, 102]. The multistate model can be fitted by maximising each transition-specific likelihood separately [101]. In other words, this means that if the distributional parameters and regression coefficients are separate for each transition, the transitions can be modelled separately [36]. In addition, different distributions can be used for each transition [22].

This approach is not possible when parameters (for example, regression coefficients or ancillary parameters) are shared across transitions. In this case, a single model can be fitted to the data in stacked format [22] and the joint likelihood from Equation 2.32 maximised [101]. Stacked format comprises of multiple entries per individual, with one entry for each transition the individual is at risk of experiencing, and more easily facilitates shared covariates across the transitions. Hazards between transitions can be proportional by constraining the ancillary parameter of the transitions (with the same distributional model) to be the same and including a covariate in the linear predictor for the transition. Regression coefficients can also be constrained to be equal across transitions [22]. Sharing parameters may be useful if there is sparse data for a transition [22].

A similar approach can be used for Cox multistate models by maximising the partial likelihood and stratifying the Cox model when separate baseline hazards and separate regression coefficients are assumed for each transition [36].

### 2.7.3 Obtaining Predictions

As with standard survival, non-, semi- and fully parametric approaches can be used to obtain metrics of interest from multistate models. In terms of non-parametric estimation, an extension to the Nelson-Aalen estimator can be used for multistate models to estimate the matrix of cumulative transition rates [103]. This can be used to calculate the non-parametric Aalen-Johansen estimate of the transition probabilities [104]. As the Aalen-Johansen estimator is a step function, expected length of stay can easily be calculated as a summation of rectangles. Non-parametric methods for multistate models are discussed in detail in Section 7.4.

As discussed in Section 2.7.2, the transition rates can be modelled relatively easily for parametric (and semi-parametric) models. The difficulty lies in obtaining transition probabilities and other important metrics from the fitted multistate model. This is because there is no longer the one-to-one relationship between the transition rate and transition probabilities that there is in the standard survival setting (Equation 2.4). The probability of moving from state  $a$  to state  $b$  depends on the overall probability of leaving state  $a$  and all transitions that leave state  $a$ .

A convenient approach for Markov models is to assume event times follow an exponential distribution, as this gives an analytical solution to the forward Kolmogorov equations [105]. Forward Kolmogorov equations are a first-order set of ordinary differential equations and the transition probabilities in a continuous time Markov model can be written in terms of these [106]. However, it may not always be feasible to assume constant transition rates. Piecewise exponential models have been suggested by Jackson [107] and can offer more flexibility when modelling the transition rates. However, a discontinuous baseline hazard function may not be biologically plausible. Another suggestion has been to model the transition rates of a Markov model with quadratic B-splines and obtain predictions by numerically

solving the forward Kolmogorov equations [106]. A more recent approach to obtaining predictions from a fitted multistate model has been suggested by Crowther and Lambert [22] that utilises a general simulation algorithm. This approach is explained in detail in Section 7.3.4.

## 2.8 Summary

This chapter introduced the key definitions, assumptions and modelling techniques for standard survival analysis, interval-censored data, IP weighting (in a survival setting) and multistate models. In particular, this chapter has focused on parametric models (including proportional hazards, Royston-Parmar and AFT models) while providing details on non- and semi-parametric approaches. The methods described here provide a foundation for the subsequent chapters of the thesis.

# Chapter 3

---

## The Impact of Naive Imputation on Interval-censored Survival Data

---

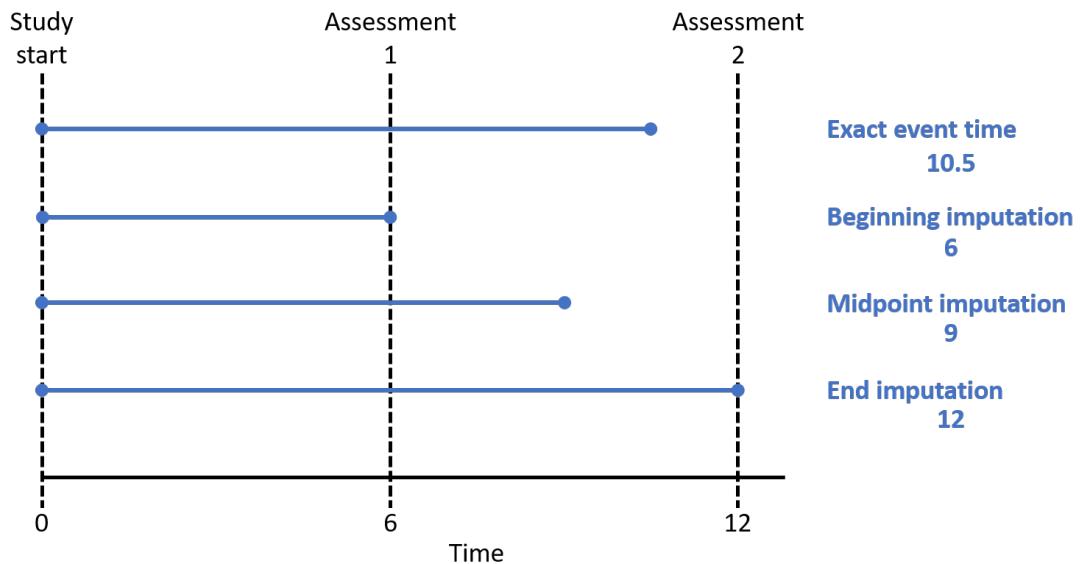
### 3.1 Outline

The aim of this chapter is to investigate the performance of naive imputation techniques on interval-censored survival data and to compare them to the likelihood-based method that appropriately accounts for interval censoring. The chapter begins with a literature review of simulation studies that evaluate the performance of the naive approaches and highlights areas still to be investigated. The naive imputation methods are then illustrated on a motivating dataset before a comprehensive simulation study addresses the gaps in the literature. The chapter finishes by providing recommendations in the context of interval-censored survival data.

### 3.2 Introduction

As discussed in Sections 1.1.2 and 2.5, naive imputation is often used to address interval-censored data, so that standard survival methods for exactly observed/right-censored data can be used. The three naive imputation techniques (beginning, mid-point and end imputation) are illustrated in Figure 3.1. Naive imputation is used despite appropriate methods that take into account interval censoring being available, see Section 2.5.2. Appropriate methods include the non-parametric Turnbull

estimator [70] and more efficient algorithms including ICM [72] and EM-ICM [73]; Finkelstein's semi-parametric estimator [74] and alternatives (Farrington's approach using generalised linear models [76] and the Cox ICM algorithm [77]); appropriate maximum likelihood estimation for parametric models and multiple imputation. One approach to handling interval-censored data is to consider them a missing data problem, for example, as is done with multiple imputation. The missing data literature generally advises against single imputation because the resulting analysis ignores the uncertainty about the unobserved value that remains despite the single imputation [108–110]. In addition, naive imputation on interval-censored data has been shown to give biased results and can lead to invalid inferences [67].



**Figure 3.1:** Illustration of the three naive imputation approaches (beginning, midpoint and end imputation)

Many authors have already flagged the use of naive methods (often end imputation) in clinical trials and longitudinal studies [111–114]. In randomised clinical trials especially, participants are often assessed at pre-scheduled, periodic follow-up visits. The date of the assessment that the (non-fatal) event is first observed is taken as the event time and treated as being observed exactly. This can be seen, for example, in oncology clinical trials [115–118] and an illustrative trial is described in the following paragraph. It can also be seen in observational studies, for example, the time to progression to AIDS in individuals with HIV [119] and the time to

the loss of various milestones (stand from supine, climb four stairs) in Duchenne Muscular Dystrophy [120].

A recent clinical trial is described here to illustrate how end imputation is applied to interval-censored data. Sugemalimab was compared to placebo after chemoradiotherapy in patients with locally advanced, unresectable, stage III non-small cell lung cancer [118]. The primary outcome was progression-free survival, according to RECIST, version 1.1, and was defined as the time from randomisation to disease progression or all-cause death, whichever occurred first [118]. Patients were assessed for progression at baseline, every 9 weeks until 12 months and then every 12 weeks until the endpoint was reached or the study ended, whichever came first. If at visit  $X$  the patient met the criteria for disease progression, the date of the assessment would be taken as the event time and would be assumed to be observed exactly. In reality, progression could have occurred at any point from just after visit  $X - 1$  to visit  $X$ . The end imputation of the progression times can be seen in the Kaplan-Meier curve in the paper (Figure 2), where the steps in the first year correspond to the 9 week (2 month) visit schedule. The figure motivates the question of whether a different assessment schedule would have influenced the results. In this example (and many others), there is the added complication that progression-free survival is a composite endpoint, where death is often observed exactly, see Zeng *et al* [121] for a discussion.

Many simulation studies have been performed to evaluate the performance of naive imputation with interval-censored data (either as the primary aim or as an additional aim). However, the results from these studies are yet to be collated and it is unclear whether there are still areas that warrant further investigation. The aim of this work, therefore, is to review and summarise the simulation literature and to identify areas that have not been explored. The second objective is to perform a comprehensive simulation study to address these gaps in the literature.

This chapter is structured as follows: Section 3.3 describes the literature review, which investigates how comprehensive the previous simulation studies were and extracts the key findings. The motivating dataset is then described and analysed

in Section 3.4. Sections 3.5 and 3.6 give the methods and results of the simulation study, respectively. The chapter finishes with a discussion and conclusion.

### 3.3 Literature Review

A literature review was performed to investigate what simulation studies had previously been performed on the topic of naive imputation on interval-censored data. Studies were included if they simulated singly interval-censored data with a univariate outcome in the standard survival setting. The study was required to investigate at least one of the three naive imputation methods and was required to estimate at least one of the following estimands: baseline model parameters, treatment effect size, median survival time and/or survival probability (at certain time points or as a function). In total, 15 studies were identified and reviewed.

The following paragraphs discuss the data generating mechanisms, estimands, methods and performance measures from the 15 studies, with the key points summarised in Table 3.1. Note that some manuscripts consisted of multiple simulation studies - only the (sub-)studies including at least one naive imputation method were considered.

In terms of data generating mechanisms, the Weibull and exponential models were the most common distributions for simulating event times, although the log-normal, log-logistic and others were also employed. Different algorithms were then employed to create interval-censored data. The most popular approach was to transform the simulated event times into interval-censored data by assuming the same regular schedule for all patients [112–114, 122–124] (results not shown for [123]). In addition, Sun and Chen [114] investigated different regular schedules for different treatment groups. Some studies [123, 125] instead/also assumed the same irregular schedule for all patients. The terms regular/irregular are used to indicate whether the time between visits was/was not consistent throughout follow-up. Three studies [122, 123, 125] also used patient-specific visits within the universal regular/irregular schedule by allowing for a small amount of variation around the

target assessment time.

Other studies assumed patient-specific schedules and this was primarily done in two ways. The first was by generating a random schedule for each subject, where the difference between visits was drawn from a Uniform distribution [126–128] or a Poisson process [129]. The second was by assuming a schedule for all patients and then allowing visits to be missed with a certain probability [80, 130] (which was possibly dependent on treatment group and possibly on event time [130]). Pantazis *et al* [130] also used patient-specific visits within the same schedule for all patients before the missed visits were applied. Williamson *et al* [131] considered a hybrid approach, generating a random schedule for each patient using the exponential distribution and then allowing visits to be missed with a probability conditional on treatment group and event time.

Finally, Odell *et al* [66] assumed a single interval and used a Bernoulli distribution to determine if the observation was interval-censored or observed exactly. Most studies considered purely interval-censored data; however, some others [127, 129] also included a mix of exactly observed and interval-censored data.

Most studies varied at least one element of the data generating mechanism, although the results were not always presented for all scenarios. A few aspects, which were of interest and/or frequently varied, are summarised in Table 3.1. Sample size was the most varied factor, followed by interval width. The treatment effect size, when included as a constant value in the data generation rather than drawn from a distribution, was the least investigated factor of those summarised (note that not all studies considered a comparison between groups).

The treatment effect size was the most frequent estimand of interest, while a small number of studies investigated the baseline model parameters (when relevant), median survival time and survival function. No study investigated the survival probability at a single time point; however, Pantazis *et al* [130] provided graphs of the true and fitted survival models, without giving quantitative details of the bias or any other performance measure for any time point.

Every study considered midpoint imputation, the majority also investigated end

imputation and four studies investigated all three imputation approaches. Harezlak and Tu [122] also considered alternate single imputation, specific to their dataset. Only three studies compared the naive approach(es) to an analysis on the data prior to interval censoring, while many studies compared the former to an appropriate method for interval censoring.

Most studies employed the Cox proportional hazards model [80, 114, 125, 126, 128–131] (this includes the use of the Cox ICM algorithm and note that this was not explicit for [125]), parametric models [66, 123, 124, 130, 131] and/or the Kaplan-Meier estimator (or version of) [112, 113, 122, 129] to analyse the simulated data. Other methods used (possibly in conjugation with another method) included survival trees [126], the (generalised) Buckley-James estimator (and extension) [127], pseudo-observations [128], the EM-ICM algorithm [128], multiple imputation [124, 129, 131], Finklestein’s method [114], the generalised Turnball estimator [129], the Nadaraya–Watson estimator [129], data augmentation [125], a resampling estimator [122] and a Monte Carlo EM algorithm for the Cox model [80].

All studies reported either the bias of the estimator, estimated value of the estimand or the mean integrated squared error (MISE). The MISE (or similar) specifically targets the survival function and is a summary measure of how close the predicted survival curve is to the true survivor function. Half of the studies reported the empirical standard error (standard deviation of the point estimates) and five gave the model-based standard errors (average of the standard errors reported in each iteration). The latter can be derived from the empirical standard error and relative error in the model-based standard error, if those were reported. Note that it was hard to decipher which standard error was reported in some studies and in these cases, the findings for the standard errors could not be summarised for that study. Støvring and Kristiansen [124] investigated the percentage increase in median standard error relative to an analysis on the exact event times (with right censoring), although they did not report the empirical standard error, model-based standard error or any results for the exact method. Finally, six studies considered coverage.

Other performance measures considered included mean squared error [66, 123], type I error rates [80, 114] and power [80, 114].

**Table 3.1:** Factors that were varied or investigated in each simulation study, denoted with a ✓

Author	Factors varied in DGM					Estimands			Methods				Performance outcomes								
	Dist	Para	Trt eff	Right cens	Int width	Samp size	Trt eff	Para est	Med	S(t) point	S(t) fct	Exact	Beg	Mid	End	IC	Bias	MISE	Emp SE	Mod SE	Cov
Dehghan [129]	✓					✓	✓					✓		✓	✓	✓	✓				
Fu [126]	✓	✓		✓	✓ <sup>1</sup>	✓ <sup>1</sup>						✓	✓	✓	✓	✓	✓		✓ <sup>2</sup>		
Gao [127]	✓						✓	✓						✓	✓	✓	✓	✓			✓
Goggins [80]			✓					✓				✓		✓	✓	✓	✓				✓
Han [128]	✓			✓				✓						✓	✓	✓	✓		✓	✓	✓
Harezlak [122]		✓	✓	✓		✓						✓	✓		✓	✓	✓	✓			
MacKenzie [123]	✓	✓ <sup>1</sup>	✓ <sup>1</sup>				✓	✓						✓	✓	✓	✓	✓		✓ <sup>3</sup>	
Odell [66]	✓			✓	✓ <sup>1</sup>	✓	✓	✓	✓ <sup>1</sup>						✓	✓	✓	✓			✓
Panageas [112]	✓				✓					✓				✓	✓	✓	✓	✓			
Pantazis [130]						✓ <sup>1</sup>	✓	✓							✓	✓	✓	✓		✓	✓
Qi [113]					✓	✓ <sup>1</sup>			✓					✓ <sup>4</sup>	✓	✓	✓	✓			
Song [125]						✓	✓								✓	✓	✓	✓		✓	✓
Størvring [124]			✓	✓	✓			✓							✓	✓	✓	✓ <sup>5</sup>			✓
Sun [114]	✓			✓		✓			✓						✓	✓	✓	✓	✓ <sup>3</sup>		✓ <sup>1</sup>
Williamson [131]									✓						✓	✓	✓	✓		✓	✓

DGM = Data generating mechanism, Dist = Event time distribution, Para = Parameters of the (baseline) event time distribution, Trt eff = Treatment effect size (constant), Right cens = Amount of right censoring, Int width = Interval width, Samp size = Sample size, Para est = (Baseline) parameter estimates, Med = Median, S(t) point = Survival probability at one/multiple time point(s), S(t) fct = Survival probability function, MISE = Mean integrated squared error, Exact = Analysis on data prior to interval censoring, Beg/Mid/End = Beginning/Midpoint/End imputation, IC = method that accounts for interval-censored data, Bias = Bias (or means to calculate it), Emp SE = Empirical standard error, Mod SE = Model-based standard error (or means to calculate it), Cov = Coverage.

<sup>1</sup>Results mentioned and/or discussed, but not shown.

<sup>2</sup>The average integrated  $L_2$  difference between the true and estimated survival curves was used [126], which is similar to MISE.

<sup>3</sup>It was unclear whether the empirical standard error or model-based standard error was reported. The former has been assumed.

<sup>4</sup>This study used one day past the previous visit for their beginning imputation (instead of the beginning of the interval).

<sup>5</sup>The median relative bias was given.

The main findings from the simulation studies were as follows:

1. All studies showed that at least one of the naive methods gave biased results (or performed poorly in terms of MISE) in at least some of the investigated scenarios.
2. Generally, midpoint imputation was the least biased naive approach (or had the lowest MISE) [113, 122, 123, 125, 127, 130] and in some scenarios gave acceptable/comparable results to the appropriate method [80, 113, 114, 123, 124, 129, 130]. However, some studies found that midpoint imputation was not consistently the least biased (or had the lowest MISE) naive approach [112, 126].
3. Some studies found that imputation methods were sensitive to the amount of right censoring, leading to more biased results (or greater MISE) as the amount of right censoring increased [124, 126] (especially as interval width increased [124]). Alternatively, Han *et al* [128] did not find a difference and Harezlak and Tu [122] found that the MISE increased slightly as the amount of right censoring increased for midpoint imputation but the reverse was true for end imputation.
4. As expected, as the interval width increased, at least some of the naive approaches gave more biased results (or greater MISE) [66, 113, 114, 124, 126, 129]; however, the magnitude of the bias varied between studies. Furthermore, Panageas *et al* [112] commented that for the median, the bias depended on the timing of the interval relative to the true median and did not necessarily increase as the interval width increased.
5. Naive approaches gave more biased results when interval widths were varied between treatment groups [114], when assessment schedules were irregular [123] and when visits were missed not at random (compared to missed at random) [130].

6. Some studies found that naive method(s) had smaller empirical standard errors compared to an appropriate method [66, 127, 128], although Song and Ma [125] and Pantazis *et al* [130] found them to be similar. Williamson *et al* [131] also found the empirical standard errors to be similar; however, the methods were not necessarily comparable.
7. Some studies found that midpoint imputation gave smaller model-based standard errors compared to an appropriate method [80, 125, 128] (although for [125] this was marginal). In addition, Williamson *et al* [131] found midpoint imputation to give artificially precise estimates (the average model-based standard error was smaller than the empirical standard error). However, the difference between the average model-based and empirical standard errors was small in some studies [125, 128, 130] (visits missed at random were considered for [130]).
8. In quite a few instances, naive method(s) had below nominal coverage [66, 114, 124, 125, 128, 130], even if this was only slightly so for some cases.

Despite the number of simulation studies investigating naive imputation methods for interval-censored data, there are still areas to be explored. In particular, none of the studies investigated survival probabilities at specified time points, which can be a useful measure of absolute risk. Despite the treatment effect being the most common estimand, it was rarely a varied factor (when included as a constant value in the data generation rather than drawn from a distribution). In addition, only three studies included a reference/oracle method that analysed the event times before interval censoring was applied. Finally, the model-based standard errors (or relative error in model-based standard errors) was only investigated in a third of the studies. The aim of the remainder of this chapter is to, therefore, perform a simulation study to address these gaps.

### 3.4 Motivating Dataset

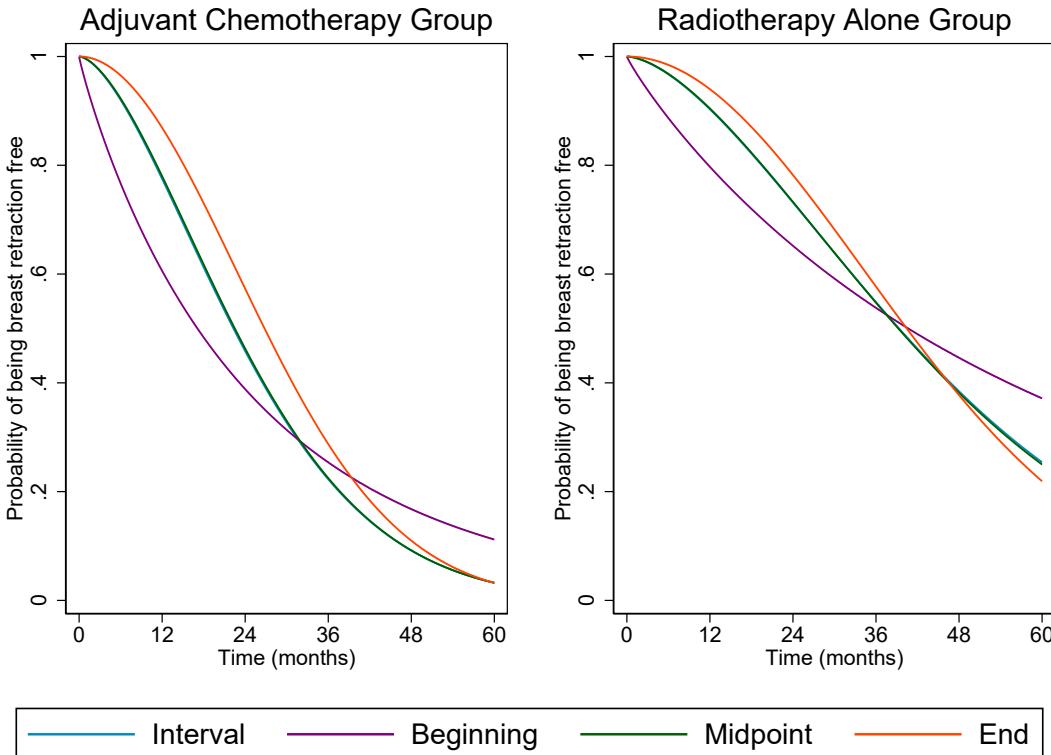
The motivating dataset is the commonly used breast cosmesis dataset, which was introduced in Section 1.3.2. The aim of the study was to compare the cosmetic effects of radiotherapy alone (control group) against radiotherapy with adjuvant chemotherapy (treatment group). The event of interest was time to the first appearance of moderate or severe breast retraction.

The dataset included 94 patients, 48 in the adjuvant chemotherapy group and 46 in the radiotherapy alone group. Patients were assessed at 6 monthly intervals for a maximum of 60 months. At the end of follow-up, 56 (59.6%) patients had experienced a moderate or severe breast retraction, 35/48 (72.9%) in the adjuvant chemotherapy group and 21/46 (45.7%) in the radiotherapy alone group.

A Weibull model was fitted to the data with treatment as a covariate using four methods. The first method appropriately accounted for interval censoring. The other three methods utilised naive imputation (beginning, midpoint and end) and then analysed the data as if it were observed exactly with right censoring. The parameter estimates are given in Table 3.2. Survival probabilities were estimated using each of the four methods and the point estimates are shown for each group in Figure 3.2. The corresponding confidence intervals are shown for each method separately in the Appendix: Figure E.1.

**Table 3.2:** Parameter estimates from a Weibull model fitted to the breast cosmesis data with the appropriate likelihood-based method for interval-censored data compared to the three naive imputation methods

Method	Log HR (SE)	$\lambda$ (SE)	$\gamma$ (SE)
Interval-censored	0.916 (0.283)	0.00185 (0.00135)	1.61 (0.193)
Beginning imputation	0.793 (0.279)	0.0233 (0.0103)	0.916 (0.111)
Midpoint imputation	0.906 (0.281)	0.00174 (0.00122)	1.63 (0.185)
End imputation	0.818 (0.278)	$4.44 \times 10^{-4}$ ( $3.66 \times 10^{-4}$ )	1.99 (0.218)



**Figure 3.2:** Survival probability estimates for the breast cosmesis data using the appropriate likelihood-based approach for interval-censored data and the three naive imputation approaches for the adjuvant chemotherapy group (left) and radiotherapy alone group (right). Note that the midpoint imputation approach overlaps the interval-censored approach

In both treatment groups, midpoint imputation gave extremely similar estimates (survival probabilities and log hazard ratio) compared to the likelihood-based approach. Both beginning and end imputation gave smaller log hazard ratios than the likelihood-based approach. Beginning imputation estimated a lower survival probability in both groups until around 32-38 months, when it then estimated a higher survival probability. End imputation gave a higher survival probability for the majority, if not the whole, of follow-up than the likelihood-based approach. The standard errors for the log hazard ratio (Table 3.2) and the confidence intervals for the survival probabilities (Figure E.1) were similar across all four approaches.

A Weibull model was chosen for demonstrational purposes and to inform the parameters of the simulation study. There may have been more appropriate model choices for these data and if the model was misspecified, it may have affected the

results. The simulation study in the next section will address this issue, as the true model for the data will be known. A Cox proportional hazards model could have been used so that no assumptions on the shape of the baseline hazard would be required. If the proportional hazards assumption was not valid, treatment could have been modelled as part of the ancillary parameter in the Weibull model, included as a time-dependent effect in a Royston-Parmar model (if more flexibility was required) or a different class of models could have been used, for example, AFT models.

## 3.5 Simulation Study Methods

The simulation study was planned according to the ADEMP structure (Aims, Data generating mechanisms, Estimands, Methods and Performance measures), proposed by Morris *et al* [4].

### 3.5.1 Aims

1. To compare the naive imputation approaches (beginning, midpoint and end imputation) with the appropriate likelihood-based method for interval-censored data in terms of bias and to investigate how this varies across different scenarios and estimands.
2. To investigate whether the model-based standard errors of the naive approaches sufficiently represent the associated uncertainty when analysing interval-censored data.
3. To evaluate the corresponding coverage.

### 3.5.2 Data Generating Mechanisms

#### General Algorithm

First,  $n_{obs}$  patients were simulated. Binary treatment variable  $X$  was drawn from a Bernoulli distribution with probability  $\pi_X$ ,  $X \sim \mathcal{B}(\pi_X)$ .

The event time  $T^*$  was assumed to follow a Weibull distribution with shape parameter  $\gamma$  and scale parameter  $\lambda$ . Let  $c_a$  be the maximum follow-up time.  $\lambda$  was chosen so the probability of being event-free (henceforth referred to as survival probability, although the event of interest need not be death) in the control group at time  $c_a$  was set to  $s_0$ ,  $S_0(c_a) = s_0$ . This was achieved by rearranging the Weibull survival function (given in Equation 2.11):

$$\lambda = -\frac{\log(s_0)}{c_a^\gamma}$$

Let  $\beta$  be the treatment effect (log hazard ratio) and  $U$  be drawn from a uniform distribution  $U \sim \mathcal{U}(0, 1)$ . The event time  $T^*$  was simulated from  $U$  as follows:

$$T^* = \left\{ \frac{-\log(U)}{\lambda \exp(\beta X)} \right\}^{1/\gamma} \quad (3.1)$$

In order to create interval-censored survival times, trajectories were constructed for each individual according to a regular schedule for all patients with patient-specific jittering. Jittering was employed to better reflect real life and was incorporated using  $\epsilon_{i,v}$ . Let  $w$  be the scheduled interval width between visits (months). The time of visit  $v$  for patient  $i$ ,  $t_{i,v}$ , was simulated as:

$$\begin{aligned} t_{i,v} &= t_{i,v-1} + w + \epsilon_{i,v} & t_{i,0} &= 0 \\ i &= \{1, 2, \dots, n_{obs}\} \\ v &= \left\{ 1, 2, \dots, \frac{c_a}{w} \right\} \\ \epsilon_{i,v} &\sim N(0, \sigma^2) & \sigma &= w/9 \end{aligned}$$

Visits  $v$  continued until the  $\frac{c_a}{w}^{th}$  visit. Note that when multiple interval widths were investigated in the study, the same simulated event time was used for each interval width. The choice of standard deviation  $\sigma$  was influenced by Mackenzie and Peng [123], who used 1/3 months for a schedule where the smallest time between visits was 3 months. The standard deviation was chosen to depend on the interval width to reflect the increased variation in visit times when visits are further apart.

The event time  $t_i^*$  for patient  $i$ , an observation of random variable  $T^*$ , was transformed into interval-censored data based on the simulated visit schedule. The corresponding interval for patient  $i$  was  $(t_{i,v-1}, t_{i,v}]$  where  $t_{i,v-1} < t_i^* \leq t_{i,v}$ . Right censoring was applied if the event time was greater than the last visit,  $t_i^* > t_{i,\frac{ca}{w}}$ . In this case, the corresponding interval was  $(t_{i,\frac{ca}{w}}, \infty)$ .

## Evaluated Scenarios

The following parameters were varied in the simulation study (factors in bold represent the motivating dataset):

- Sample size:  $n_{obs} = \{100, 500\}$
- Shape parameter of the Weibull model:  $\gamma = \{0.25, 0.7, 1.6, 3.0\}$
- Survival probability in the control group at time  $c_a$ :  $s_0 = \{0.25, 0.75\}$
- Treatment effect size:  $\beta = \log(\text{HR}) = \{0.14, 0.92\}$
- Interval width:  $w = \{3, 6, 12\}$  months

Four shape parameters were chosen to represent rapidly/slowly increasing/decreasing hazard rates. For each shape parameter, two scale parameters,  $\lambda$ , were chosen to represent a common and less common event in the control group. The resulting  $\lambda$  values are given in Table 3.3. The various hazard functions in the control group from the data generating mechanism for  $s_0 = 0.25$  are shown in Figure 3.3, left panel. The right panel gives the survival function in the control group for all 8 (baseline) Weibull data generating mechanisms. Two treatment effect sizes were chosen to represent a strong and weak (harmful) treatment effect.

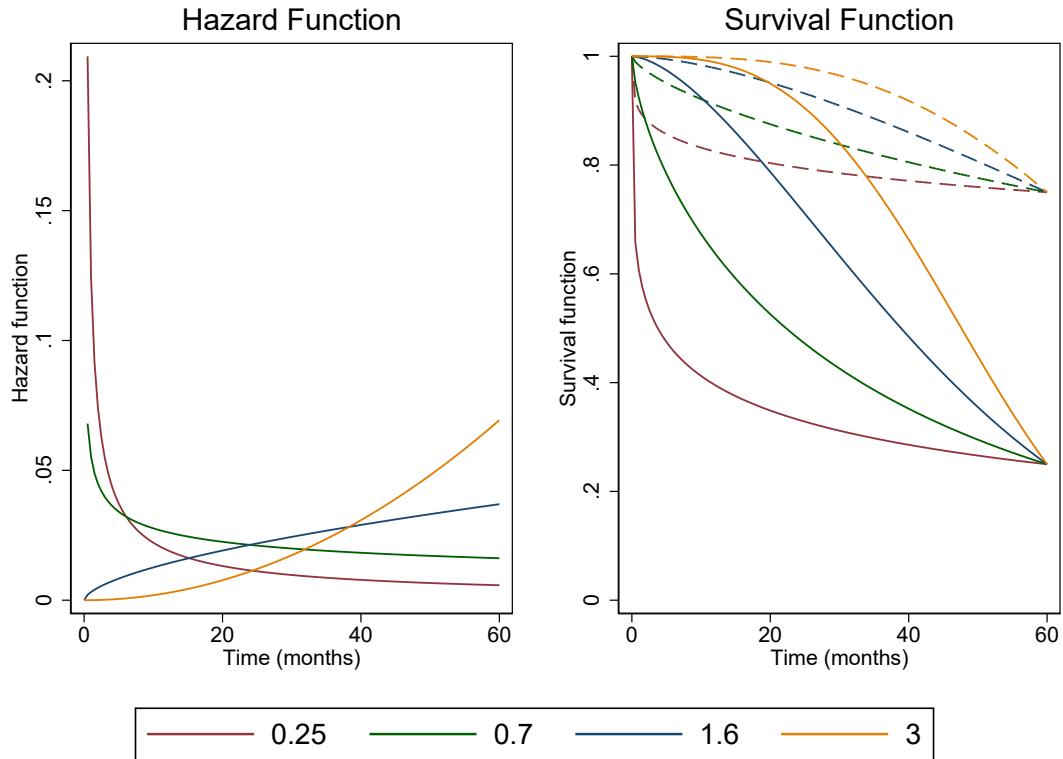
Three interval widths were chosen: 3 (similar to assessments in some oncology clinical trials), 6 (as per the motivating dataset) and 12 months (similar to assessments in observational studies with annual follow-up). The corresponding standard deviations mean that most simulated patients will have had their visit within 0.5, 1 and 2 months of the scheduled visit time for interval widths 3, 6 and 12 months,

respectively. Finally, an additional sample size of 500 was investigated to represent a possible (small) phase III oncology trial.

This led to a total of  $2 \times 4 \times 2 \times 2 \times 3 = 96$  scenarios.

**Table 3.3:**  $\lambda$  values for the data generating mechanism, rounded to 3 significant figures (the exact values were used in the simulation study)

$\gamma$	$s_0 = 0.25$	$s_0 = 0.75$
0.25	0.498	0.103
0.7	0.0789	0.0164
1.6	0.00198	$4.11 \times 10^{-4}$
3	$6.42 \times 10^{-6}$	$1.33 \times 10^{-6}$



**Figure 3.3:** Left panel: The hazard function in the control group from the data generating mechanism for the four  $\gamma$  values, where  $c_a = 60$  months and  $S_0(60) = 0.25$  (the survival probability at 60 months is 25%). Right panel: The survival function in the control group from the data generating mechanism for the four  $\gamma$  values, where  $S_0(60) = 0.25$  is shown by the solid lines and  $S_0(60) = 0.75$  is shown by the dashed lines

## Simulation Parameters

The simulation parameters were chosen to reflect the motivating dataset.  $\pi_X$  was set as 0.5 to reflect an equal probability of being allocated each treatment. The maximum follow-up time was set to 60 months,  $c_a = 60$ . This resulted in 20, 10 and 5 follow-up visits for interval widths 3, 6 and 12 months, respectively.

### 3.5.3 Estimands

The following estimands were of interest:

1. The log hazard ratio  $\beta$
2. The survival probability at yearly time points (12, 24, 36 and 48 months) in the control group
3. The survival function estimated across the whole follow-up time in the control group (the estimand for MISE)

### 3.5.4 Methods

For all methods, if the event time  $t_i^*$  for individual  $i$  occurred after the last visit ( $t_i^* > t_{i,\frac{c_a}{w}}$ ), right censoring was applied. Otherwise, let the event time  $t_i^*$  fall in interval  $(t_{i,v-1}, t_{i,v}]$ . The following five methods were employed, all utilising a Weibull model:

1. **Reference method (Exact):** The survival time was set as the event time ( $t_i = t_i^*$ ) and a Weibull model for exactly observed/right-censored data was fitted. This method would not be available in practice.
2. **Beginning imputation (Beg):** The survival time was imputed as the beginning time of the interval ( $t_i = t_{i,v-1}$ ) and a Weibull model for exactly observed/right-censored data was fitted. Note, if the event occurred in the first interval  $(0, t_{i,1}]$ , then 1 day (1/30 months) was imputed instead of 0 ( $t_i = 1/30$ ).

3. **Midpoint imputation (Mid):** The survival time was imputed as the midpoint of the interval ( $t_i = \frac{t_{i,v-1}+t_{i,v}}{2}$ ) and a Weibull model for exactly observed/right-censored data was fitted.
4. **End imputation (End):** The survival time was imputed as the end time of the interval ( $t_i = t_{i,v}$ ) and a Weibull model for exactly observed/right-censored data was fitted.
5. **Interval censoring method (IC):** The event time was taken to fall in the interval  $[t_{i,v-1}, t_{i,v}]$ . A Weibull model was fitted using the likelihood-based method, taking interval censoring into account.

### 3.5.5 Performance Measures

The following notation will be used, based on Table 2 from Morris *et al* [4]:

**Table 3.4:** Description of simulation study notation, a subset of Table 2 from Morris *et al* [4]

$\theta$	An estimand, also the true value of the estimand
$n_{sim}$	Number of iterations, iteration sample size
$\hat{\theta}$	The estimator of $\theta$
$\hat{\theta}_i$	The estimate of $\theta$ from the $i^{th}$ iteration
$\bar{\theta}$	The mean of $\hat{\theta}_i$ across iterations
$\text{Var}(\hat{\theta})$	The true variance of $\hat{\theta}$
$\widehat{\text{Var}}(\hat{\theta}_i)$	An estimate of $\text{Var}(\hat{\theta})$ from the $i^{th}$ iteration

The empirical standard error (EmpSE) is defined as  $\sqrt{\text{Var}(\hat{\theta})}$  while the average model-based standard error (ModSE) is defined as  $\sqrt{\text{E}\{\widehat{\text{Var}}(\hat{\theta})\}}$  [4]. The EmpSE estimates the standard deviation of  $\hat{\theta}_i$  over the  $n_{sim}$  iterations [4]. Alternatively, the ModSE takes the average of the variance estimated in each iteration  $i$  and then takes the square root.

The following performance measures were analysed for each aim for the first two estimands (log hazard ratio and survival probabilities at yearly time points):

1. **Aim 1:** The bias of the point estimates:  $E(\hat{\theta}) - \theta$  and corresponding Monte Carlo Standard Error (MCSE), calculated as [4]:

$$MCSE = \sqrt{\frac{1}{n_{sim}(n_{sim}-1)} \sum_{i=1}^{n_{sim}} (\hat{\theta}_i - \bar{\theta})^2} = \sqrt{\frac{\text{Var}(\hat{\theta})}{n_{sim}}} \quad (3.2)$$

2. **Aim 2:** The relative percentage error in ModSE:  $100\left(\frac{\text{ModSE}}{\text{EmpSE}} - 1\right)$  and corresponding MCSE, calculated as [4]:

$$MCSE = 100 \left( \frac{\widehat{\text{ModSE}}}{\widehat{\text{EmpSE}}} \right) \sqrt{\frac{\widehat{\text{Var}}\{\widehat{\text{Var}}(\hat{\theta})\}}{4n_{sim} \times \widehat{\text{ModSE}}^4} + \frac{1}{2(n_{sim}-1)}} \quad (3.3)$$

3. **Aim 3:** The coverage:  $P(\hat{\theta}_{low} \leq \theta \leq \hat{\theta}_{upp})$ , where the estimated confidence interval for  $\hat{\theta}$  is  $(\hat{\theta}_{low}, \hat{\theta}_{upp})$ . Note that confidence intervals were calculated on the  $\log(-\log(\cdot))$  scale for the survival probabilities at yearly time points to ensure the confidence intervals (on the original scale) would lie in the  $[0, 1]$  boundary. Let  $\widehat{\text{Cover}}$  be the estimated coverage, then the corresponding MCSE is calculated as [4]:

$$MCSE = \sqrt{\frac{\widehat{\text{Cover}} \times (1 - \widehat{\text{Cover}})}{n_{sim}}}$$

Note that  $\widehat{\text{Var}}\{\widehat{\text{Var}}(\hat{\theta})\}$  can be interpreted as the estimated variance of the (model-based) variance estimates from each iteration  $i$ :  $\widehat{\text{Var}}(\hat{\theta}_i)$ .

In addition, the MISE was used to address Aim 1 for the third estimand (the survival function). The MISE is a summary measure of bias across the follow-up time and is defined as the following, where  $S(u)$  is the true survival probability at time  $u$  and  $\widehat{S}(u)$  is the predicted survival probability at time  $u$ :

$$\text{MISE} = E \left[ \int_0^t \left\{ S(u) - \widehat{S}(u) \right\}^2 du \right]$$

An upper limit of  $t = c_a$  was chosen to ensure the definition was consistent across iterations (instead of choosing the maximum event time in each iteration for example). MISE was approximated using numerical integration, calculated at 1000 intervals.

### Iteration Sample Size

Bias was chosen as the primary performance measure of interest. An iteration sample size calculation was performed to ensure the bias (in Aim 1) was estimated to an acceptable degree of precision [4]. Equation 3.2 can be rearranged to give the iteration sample size for a given value of MCSE:

$$n_{sim} = \frac{1}{MCSE^2 (n_{sim} - 1)} \sum_{i=1}^{n_{sim}} (\hat{\theta}_i - \bar{\theta})^2 = \frac{\text{Var}(\hat{\theta})}{MCSE^2} \quad (3.4)$$

A maximum MCSE value for bias of 0.005 months was deemed acceptable. Estimates of  $\text{Var}(\hat{\theta})$  (or EmpSE<sup>2</sup>) were calculated by performing the simulation with 100 iterations. Estimates of  $\text{Var}(\hat{\theta})$  were for each of the first two estimands, for each method and for each of the 96 scenarios. The maximum over the scenarios and methods was taken for each estimand and the iteration sample size for the main study was chosen as the maximum across the estimands. The results are shown in Table 3.5. The iteration sample size was determined to be  $n_{sim} = 9200$  for each scenario.

### 3.5.6 Software

The simulations were performed using **Stata**, version 15.1, and the analysis of the simulated datasets was performed in **Stata**, version 17.0. A later version of **Stata** was used for the analysis of the simulated datasets as this work was re-ran at the end of the PhD to provide correctly formatted tables and figures for the thesis and at this point version 17.0 was being used.

The **survsim** package [132] was used to simulate the event times ( $T^*$ ) from a Weibull distribution, as shown in Equation 3.1. Methods 1-4 in Section 3.5.4

(Exact, Beg, Mid and End) were implemented using `streg`, while the IC method was performed using `stintreg`. The `simsum` package [133] was used to calculate the performance measures in Section 3.5.5.

Data were simulated using the 64-bit Mersenne twister for random number generation. The input seed was 86291 for the iteration sample size calculation ( $n_{sim} = 100$ ) and 4387562 for the main simulation study ( $n_{sim} = 9200$ ).

**Table 3.5:** Estimated maximum variance for each of the first two estimands from the preliminary simulation with 100 iterations, corresponding  $n_{sim}$  required so that the MCSE for bias is less than or equal 0.005 and the estimated maximum MCSE when  $n_{sim} = 9200$

Estimand	Maximum Var $(\hat{\theta})$	$n_{sim}$ needed for MCSE $\leq 0.005$	Maximum MCSE if $n_{sim} = 9200$
<b>Log HR</b>	<b>0.23</b>	<b>9200</b>	<b>0.005</b>
$S_0(12)$	0.034	1360	0.0019
$S_0(24)$	0.046	1840	0.0022
$S_0(36)$	0.052	2080	0.0024
$S_0(48)$	0.058	2320	0.0025

## 3.6 Simulation Study Results

### 3.6.1 Exploratory Analysis

#### Convergence Issues

Of the 4,416,000 analyses (96 scenarios  $\times$  9200 iterations  $\times$  5 methods), 8 did not converge for the IC method and therefore the estimands could not be estimated. 7/8 cases resulted in an error, while the other case reached the maximum number of iterations in the maximisation algorithm without converging. All 8 cases had shape parameter ( $\gamma$ ) as 0.25, survival probability at 60 months as 75%, log hazard ratio as 0.14 and sample size as 100. The interval width included some 6 month and mostly 12 month intervals. In all non-converged cases, there were only left-censored (interval-censored in the first interval) and right-censored event times in both treatment groups.

For consistency and due to the small numbers, all results (all methods and estimands) from any dataset with only left-censored and right-censored event times in both treatment groups were removed. This led to the removal of 21 datasets, which meant 105 (21 datasets  $\times$  5 methods) analyses. Note that (as mentioned in Section 3.5.2) the same simulated event time is used to create three datasets, one for each of the three interval widths. The 21 datasets comprised of 7 sets of event times  $\times$  3 interval widths, where each set of event times contributed to at least one convergence issue (one set contributed to a convergence issue for both the 6 and 12 month interval).

## Extreme Values

There did not appear to be any extreme values for the survival probability at yearly time points estimand. However, exploratory graphs revealed 68 possible outliers for the log hazard ratio estimand, where the model-based standard errors were considerably large. All cases had shape parameter ( $\gamma$ ) as 0.25, survival probability at 60 months as 25%, log hazard ratio as 0.92 and sample size as 100. The interval width was mostly 12 months with a few 6 months. The extreme model-based standard errors were produced from the IC method when all of the events in the treatment group were left-censored.

For consistency and due to the small numbers, all results (all methods and estimands) where all events in the treatment group were left-censored were removed. This led to the removal of 186 datasets, which meant 930 (186 datasets  $\times$  5 methods) analyses. The 186 datasets comprised of 62 sets of event times  $\times$  3 interval widths, where each set of event times contributed to at least one extreme value (six sets contributed to an extreme value for both the 6 and 12 month interval).

In total, this led to an exclusion of a very small number of datasets:  $21 + 186 = 207$  out of  $96 \text{ scenarios} \times 9200 \text{ iterations} = 883,200$  total datasets.

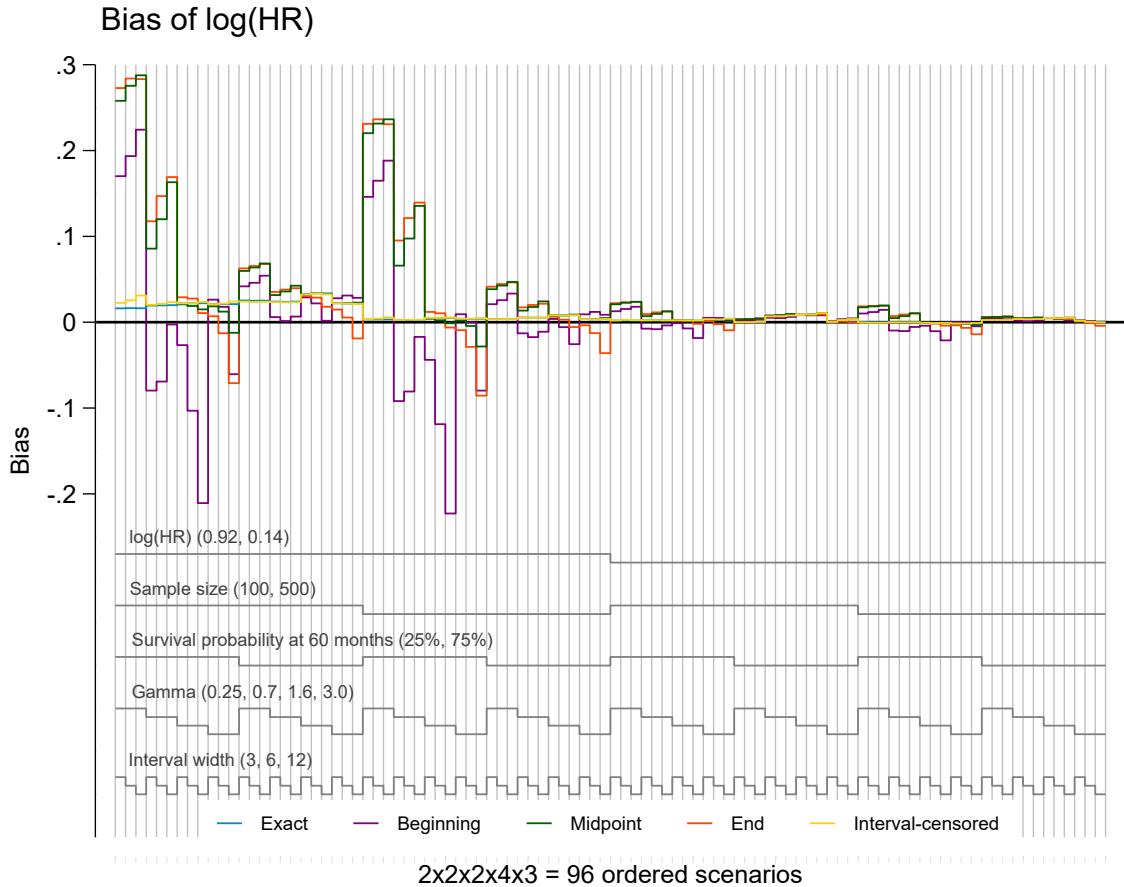
### 3.6.2 Main Analysis

The maximum estimated MCSE for the bias was 0.0045, 0.0008, 0.0008, 0.0009 and 0.0008 for the marginal log hazard ratio and survival probability in the control group at 12, 24, 36 and 48 months, respectively. These were below the maximum acceptable MCSE threshold of 0.005, as specified in Section 3.5.5, and suggests that the iteration sample size was sufficient. The average MCSE across the scenarios and methods was 0.0024, 0.0002, 0.0003, 0.0004 and 0.0004, respectively. The maximum MCSE was 2.05 percentage points (survival probability at 12 months) and 0.52 (all estimands) for the relative percentage error in model-based standard errors and coverage, respectively.

#### Aim 1: Bias

The bias of the log hazard ratio is shown for each of the 96 scenarios in Figure 3.4 using a nested loop plot. In nearly all scenarios, the IC method performed similarly to the exact method and both appeared to give unbiased estimates, except when the log hazard ratio was 0.92 and the sample size was 100. All naive methods exhibited some level of bias in at least some of the scenarios. Generally, a larger hazard ratio and smaller survival probability at 60 months gave more noticeably biased results for the naive methods, while decreasing the sample size and increasing the interval width gave slightly more biased results.

The impact of  $\gamma$  on bias varied for the naive approaches. Generally,  $\gamma = 0.25$  led to the greatest bias, with the naive methods overestimating the true log hazard ratio. As  $\gamma$  increased, the bias decreased for the midpoint and end imputation approaches, with it stabilising for midpoint imputation when  $\gamma = 1.6$ . The absolute bias increased for end imputation when  $\gamma = 3$ , although in the opposite direction. Beginning imputation underestimated the true value of the log hazard ratio for less extreme  $\gamma$  values (0.7 and 1.6), giving greater absolute bias for  $\gamma = 1.6$ , and then gave relatively unbiased estimates for  $\gamma = 3$ .

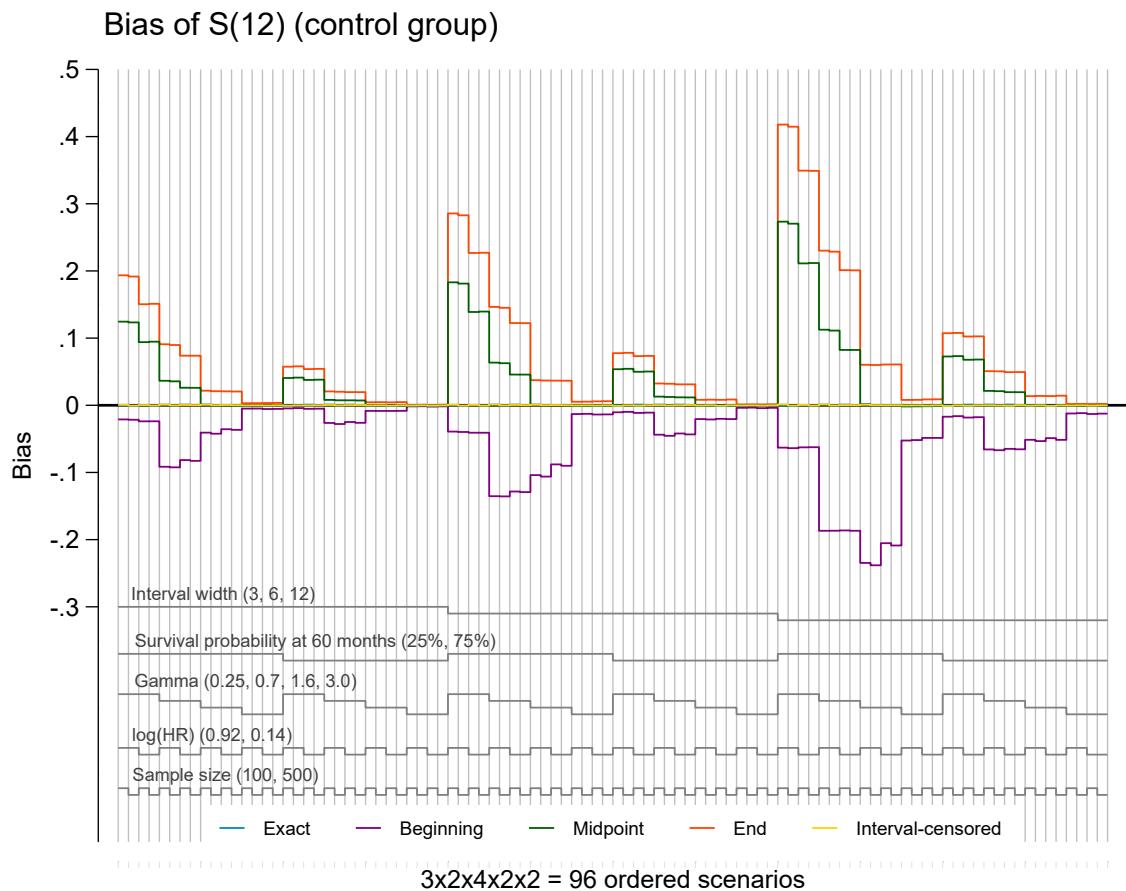


**Figure 3.4:** Nested loop plot showing the bias of the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. For example, the fifth step represents a log hazard ratio of 0.92, sample size of 100, survival probability at 60 months of 25%,  $\gamma$  value of 0.7 and interval width of 6 months. Note that the IC method often overlaps the exact method

Generally, midpoint imputation was less biased than end imputation, although in some cases they gave similar results. In some scenarios (for example, log hazard ratio of 0.14 and 75% survival probability at 60 months), they appeared relatively unbiased. Beginning imputation gave varying levels of bias, and was sometimes the least biased naive approach (for example,  $\gamma = 0.25$ ) and sometimes the most biased naive approach (for example,  $\gamma = 1.6$ ). The MCSE could not be shown on the graph, but is given (along with the point estimates) for selected scenarios in the Appendix: Table E.1.

The bias of the survival probability at 12 months (in the control group) is shown in Figure 3.5. Similarly to the log hazard ratio, the IC method performed similarly to

the exact method and both appeared to be unbiased. Alternatively, there were many scenarios where the naive methods appeared to be biased, with beginning imputation underestimating the true value and the other two naive methods overestimating the true value. Midpoint imputation was the consistently least biased naive approach and was comparable to the IC method in about half of the scenarios (when  $\gamma > 1$ ). However, there were still scenarios, like with the log hazard ratio, where beginning imputation was the least biased estimator (when  $\gamma = 0.25$ ).



**Figure 3.5:** Nested loop plot showing the bias of the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

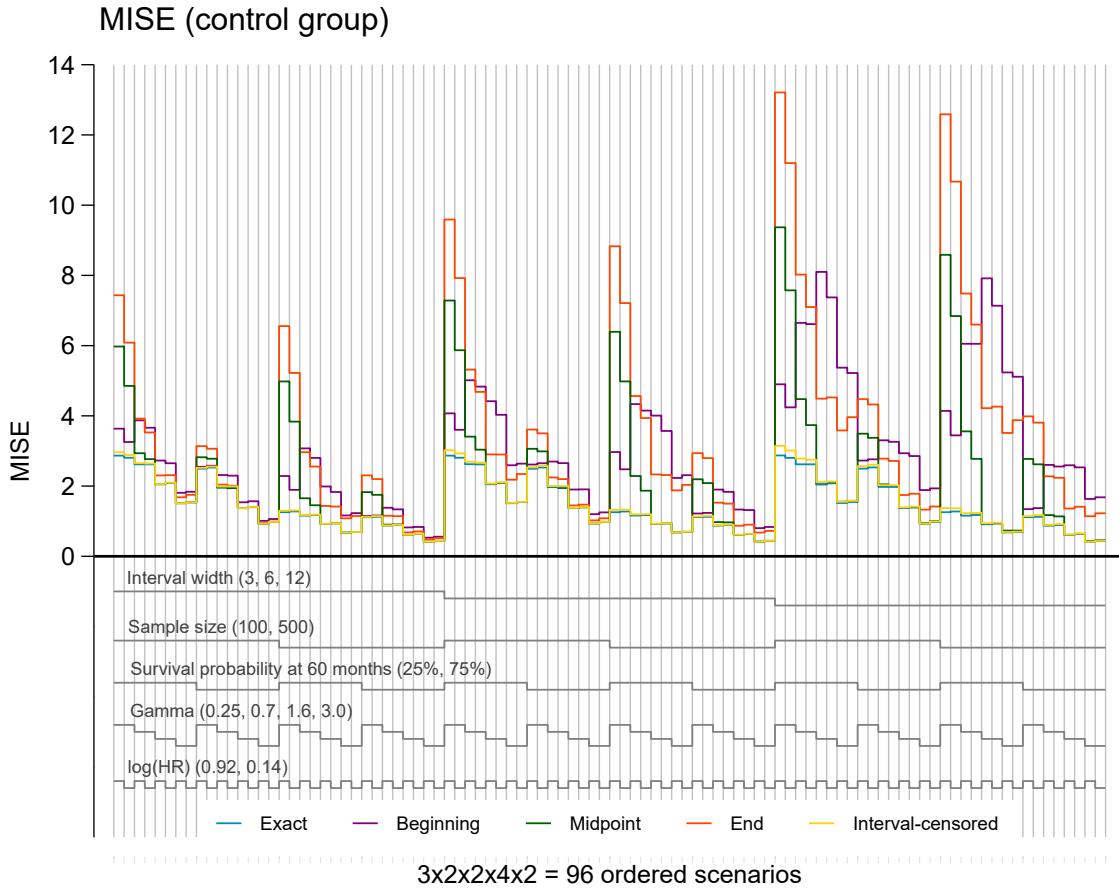
Generally, a smaller survival probability at 60 months and larger interval width gave more noticeably biased results for the naive methods. In some cases, an increase in the log hazard ratio increased the bias for the naive method, while the sample size appeared to have relatively little impact on the bias. The bias decreased as  $\gamma$

increased for both midpoint and end imputation (the bias was similar for midpoint imputation when  $\gamma > 1$ ). The absolute bias was greatest for beginning imputation when  $\gamma$  was less extreme (0.7 and 1.6).

Figures E.2, E.3 and E.4 in the Appendix show the bias of the survival probability at 24, 36 and 48 months (in the control group), respectively. As the time point of interest increased, similar trends could be observed but the absolute bias decreased. By 36 months, midpoint imputation appeared to be unbiased in many scenarios. As the time point of interest increased, the relationship between  $\gamma$  and bias differed. For example, at 48 months, the absolute bias increased as  $\gamma$  increased for beginning and end imputation (with  $\gamma = 0.25$  being a slight exception). The MCSE is given for selected scenarios for the survival probabilities at times 12 and 48 months in the Appendix: Table E.4.

The MISE (in the control group) is shown in Figure 3.6. Similarly to the other estimands, the IC method had similar MISE values to the exact method. In many scenarios, all naive approaches had greater MISE than the exact and IC methods. Midpoint imputation was the overall best performing naive method and gave a similar MISE value to the IC method in some cases.

Similar trends to those in the survival probability at specific time points estimand were observed, for example, MISE increased as the interval width increased, survival probability at 60 months decreased, the log hazard ratio increased (in some cases) and (generally)  $\gamma$  decreased. The main exception was that an increased sample size appeared to noticeably decrease the MISE for all methods.

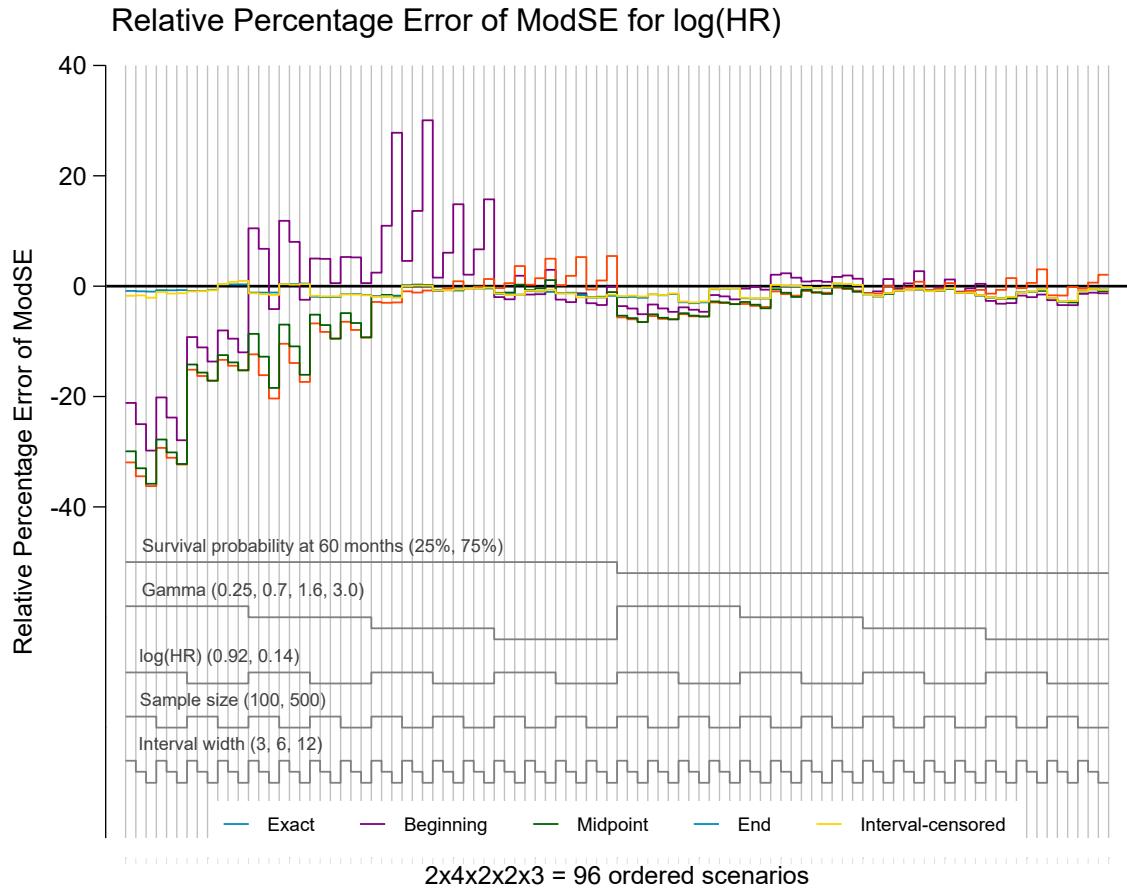


**Figure 3.6:** Nested loop plot showing the MISE (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

### Aim 2: Relative Percentage Error In Model-Based Standard Errors

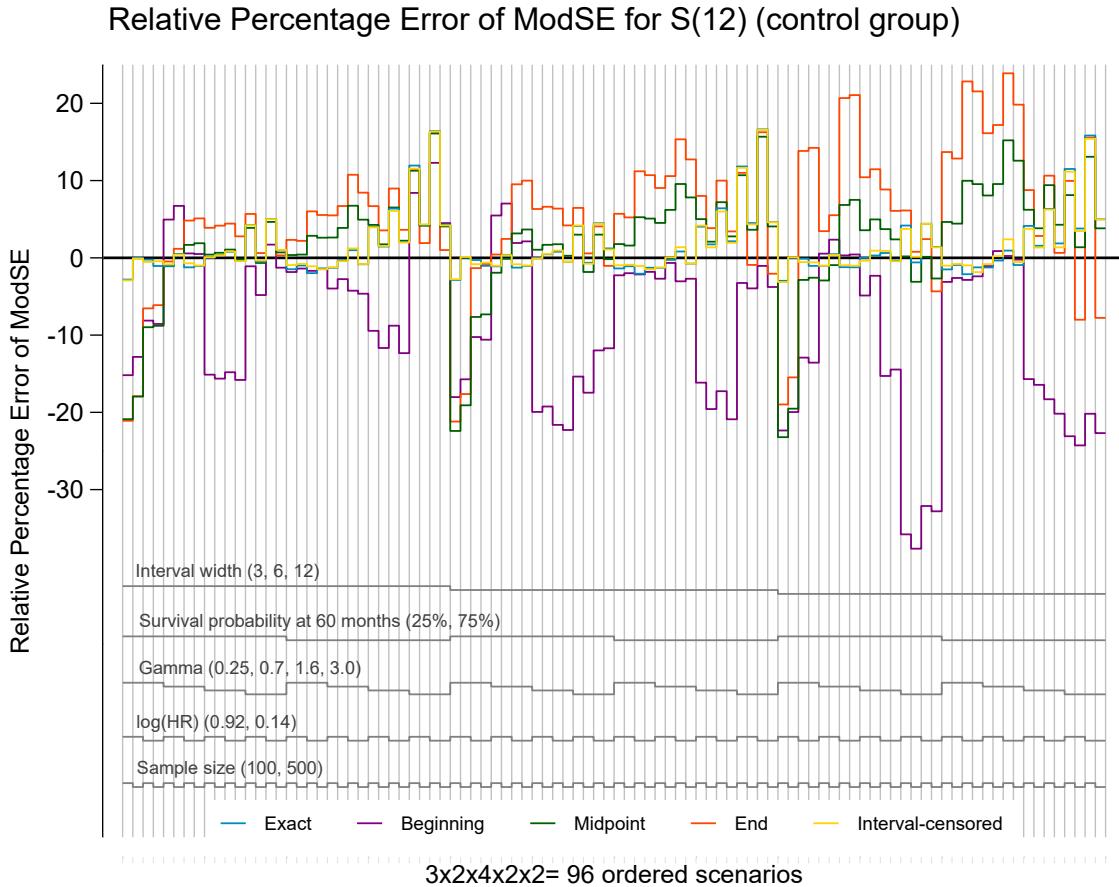
The relative percentage error of the model-based standard errors for the log hazard ratio is shown in Figure 3.7. As before, the IC method performed similarly to the exact method and both gave relative percentage errors close to 0 in most cases. Similar trends with  $\gamma$  and the survival probability at 60 months were seen here with the naive methods as were seen with bias. For example, when  $\gamma = 0.25$  and the survival probability at 60 months was 25%, all naive methods showed a large, negative relative percentage error, indicating that the model-based standard errors were considerably smaller than the standard deviation of the point estimates. Similarly to bias, an increase in the interval width, log hazard ratio and sample size led to an increase in absolute relative error for the naive methods. As before,

midpoint imputation appeared to generally have smaller absolute relative error than end imputation, with beginning imputation exhibiting varying levels of relative error across the scenarios. The MCSE is given in the Appendix: Table E.2 for selected scenarios.



**Figure 3.7:** Nested loop plot showing the relative percentage error in model-based standard errors for the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

The relative percentage error in the model-based standard errors for the survival probability at 12 months (in the control group) is shown in Figure 3.8. As before, the IC method performed similarly to the exact method, although the relative error was not always close to 0. The absolute relative error for these methods was greater as the survival probability at 60 months increased, sample size decreased and  $\gamma$  decreased.



**Figure 3.8:** Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

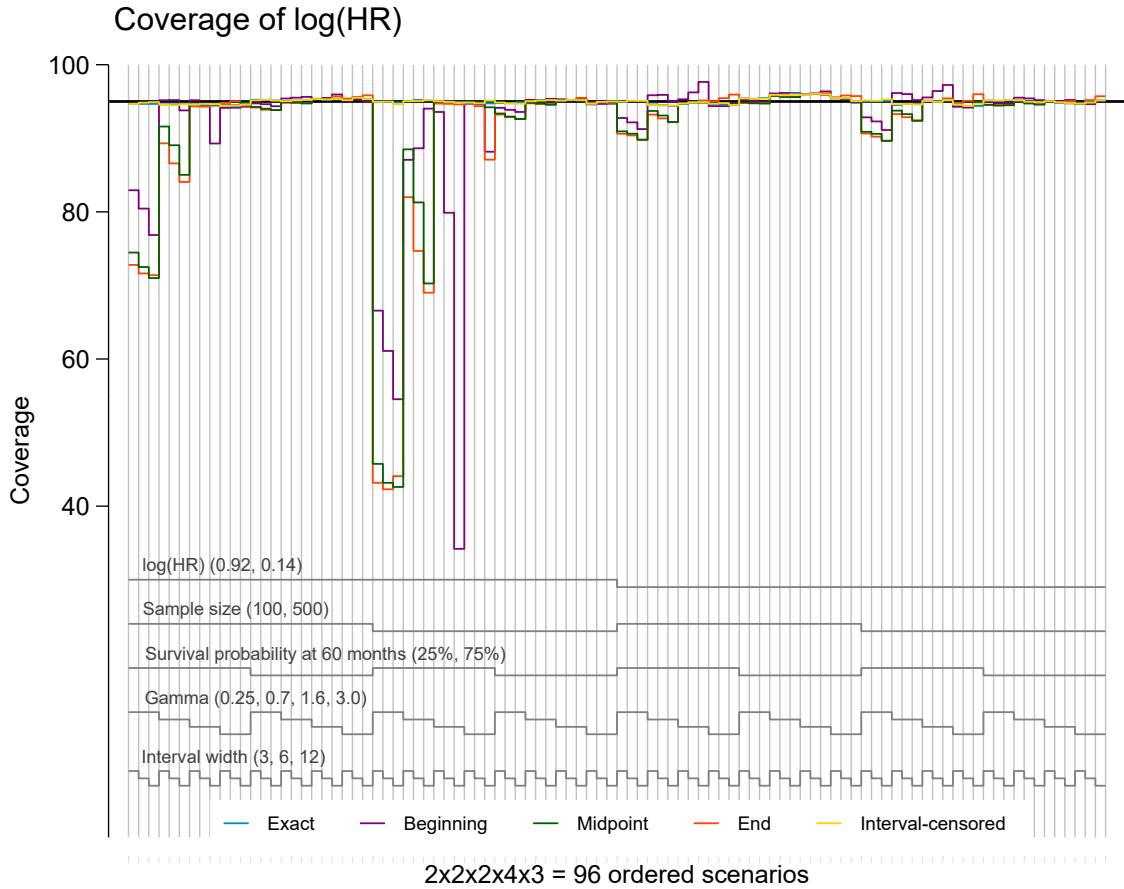
There were many scenarios where the naive methods exhibited large, absolute relative errors. Generally, the model-based standard errors from beginning/midpoint/end imputation tended to under-/over-/overestimate the empirical standard error. Midpoint imputation generally gave smaller absolute relative errors than end imputation, with beginning imputation giving both smaller and larger relative errors across the scenarios. For the naive methods, the absolute relative error increased as the interval width increased and the survival probability at 60 months increased (midpoint and end imputation).  $\gamma$  continued to affect the naive methods in different ways.

Figures E.5, E.6 and E.7 in the Appendix show the relative percentage error of the model-based standard errors for the survival probability at 24, 36 and 48 months

(in the control group), respectively. Generally, the absolute relative error decreased for all methods as the time point increased. By 36 months, more distinct trends were apparent. Similarly to the log hazard ratio, when  $\gamma = 0.25$  and the survival probability at 60 months was 25%, all naive methods showed a large, negative relative percentage error. As  $\gamma$  increased, the absolute relative error decreased for the midpoint and end imputation approaches. For beginning imputation, the trend with  $\gamma$  varied with interval width. The MCSE is given for selected scenarios for the survival probabilities at times 12 and 48 months in the Appendix: Table E.5.

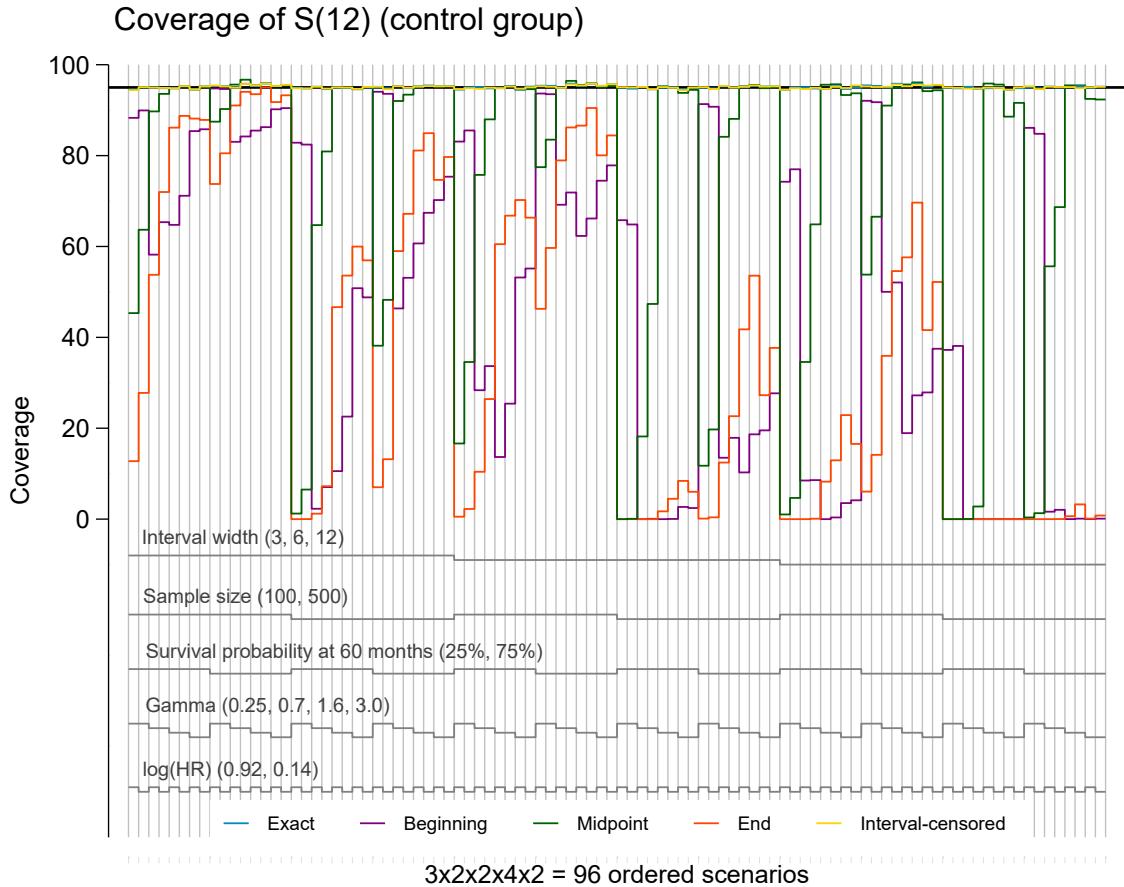
### Aim 3: Coverage

The coverage of the log hazard ratio is shown in Figure 3.9. As before, the IC method performed similarly to the exact method and both were close to the nominal level of 95% in most cases. There were many scenarios where the coverage of the naive methods were near the nominal 95% level; however, there were some instances where they were considerably below this level. This was unsurprising as under-coverage is expected when estimates are biased and/or the model-based standard error is less than the empirical standard error [4]. In particular, when the log hazard ratio was 0.92 and the survival probability at 60 months was 25%, the naive methods exhibited severe under-coverage. This was amplified when the sample size increased to 500. Similar trends with  $\gamma$  and the naive methods were seen here as were seen with bias and relative error. The MCSE is given in the Appendix: Table E.3 for selected scenarios.



**Figure 3.9:** Nested loop plot showing the coverage of the log hazard ratio across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

The coverage of the survival probability at 12 months (in the control group) is shown in Figure 3.10. As with the log hazard ratio, the IC method performed similarly to the exact method and both were close to the nominal level of 95% in most cases. However, unlike the log hazard ratio, the naive methods frequently exhibited severe under-coverage. As discussed above, under-coverage was expected for some scenarios. The ranking of the naive methods was similar to before with this estimand. The naive methods had worse coverage as the interval width increased, sample size increased, survival probability at 60 months decreased and log hazard ratio increased (in some cases). The relationship with  $\gamma$  was similar to before.



**Figure 3.10:** Nested loop plot showing the coverage of the survival probability at 12 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

Figures E.8, E.9 and E.10 in the Appendix show the coverage of the survival probability at 24, 36 and 48 months (in the control group), respectively. As the time point increased, the performance of the naive methods improved and there were more cases where the naive methods achieved the nominal level of coverage. However, there were still a number of cases at 48 months where there was severe under-coverage. The trends were similar as for survival at 12 months. The MCSE is given for selected scenarios for the survival probabilities at times 12 and 48 months in the Appendix: Table E.6.

### 3.7 Discussion

This chapter began with a literature review to investigate what simulation studies had been previously performed on the topic of naive imputation on interval-censored data. 15 studies were included and, although many areas had been covered in the literature, no study had investigated the survival probability at specific time points and few had evaluated the model-based standard errors or included a reference method (analysis on the event times before interval censoring had been applied). This motivated the need for a further simulation study.

In the simulation study performed in this chapter, the appropriate likelihood-based method for interval-censored data gave consistent results with the reference method and both performed well overall (no/small bias, no/small relative error in the model-based standard errors and coverage close to the nominal 95% level). As found in the literature review, all naive methods performed poorly in at least some of the scenarios, while there were also instances where their performance was acceptable. Generally, midpoint imputation performed better than endpoint imputation, with beginning imputation performing the best and worst out of the naive methods in different scenarios. The literature review also found that, generally, midpoint imputation was the least biased naive approach (or had the lowest MISE). Note that beginning imputation was not investigated in many of the reviewed studies.

As observed by Williamson *et al* [131], there were cases when the naive methods had artificially precise standard errors. In particular, this occurred when  $\gamma$  was small (0.25) and the survival probability at 60 months was 25% (especially for the later time points for the survival probability estimand). However, there were also scenarios where the relative error was reasonable, which was found in other studies [125, 128, 130]. In terms of coverage, there were instances where the naive methods appeared to give both acceptable and unacceptable levels of coverage for the log hazard ratio. However, for the survival probability estimand (all time points) the naive methods resulted in severe under-coverage for the majority of scenarios. Coverage below the nominal levels had been noted in other simulation studies [66,

114, 124, 125, 128, 130], even if this was only slight in some cases.

Generally, increasing the interval width, decreasing the survival probability at 60 months and increasing the treatment effect resulted in poorer performances for the naive methods. The effect of increasing the interval width had been previously observed [66, 113, 114, 124, 126, 129]. However, with regards to right censoring, other studies had found that increasing the amount of right censoring had resulted in greater bias (rather than less) for the naive methods [124, 126]. Although, it should be noted that not all studies investigating right censoring came to this conclusion [122, 128]. Increasing the sample size considerably reduced the coverage for both estimands, as was seen in other studies [66, 124, 125]. The impact of  $\gamma$  varied across the imputation methods. In particular, midpoint and end imputation performed worse when  $\gamma$  was 0.25 and Odell *et al* [66] also found that midpoint imputation gave the largest absolute bias for the log hazard ratio when the hazard function was rapidly decreasing. Finally, as the time point increased for the survival probability estimand, the performance of all naive methods generally improved.

Although a range of scenarios was investigated in the simulation study, many extensions could have been considered. Firstly, the choice of a Weibull data generating distribution is restrictive, as it can only facilitate a monotonic hazard function. Distributions such as the log-normal or generalised gamma would have been able to model a more complex hazard function with a turning point. Furthermore, fractional polynomials, mixture models or Royston-Parmar models could have been used to simulate more biologically plausible data. However, this would have raised the question of what model to fit to the data and the topics of model selection and model misspecification would have needed investigation.

Secondly, only a single covariate was considered, which was binary and time-independent. The study could have been extended to look at a number of covariates, including continuous, time-dependent and/or time-varying covariates.

A third extension could have been to vary the assessment schedule. An irregular schedule for all patients could have been investigated. In practice, this can occur if patients are expected to experience the event of interest quickly and are therefore

seen more frequently at the beginning of the study. Alternatively, patient specific schedules could have been employed by allowing individuals to miss visits, see Pantazis *et al* [130] and Williamson *et al* [131]. The visits could be missed at random (for example, an individual may forget to attend a follow-up visit), depend on the treatment (for example, patients on treatment may not be well enough to attend the visit due to adverse events) and/or depend on event time. In addition, patients in the control group could have additional visits, as they may unexpectedly be seen more often due to quicker, symptomatic progression.

Finally, this simulation study investigated singly interval-censored data with a univariate outcome. Doubly interval-censored data, interval-censored covariates, multivariate outcome data and a mixture of interval-censored and exactly observed data were outside the scope of this study. The latter may arise, for example, with the endpoint of progression-free survival where progression is interval-censored and death is observed exactly. In addition, the simulation study focused on the standard survival setting; more complex survival data with interval censoring, for example, competing risks, multistate survival models and longitudinal data with survival outcomes (joint modelling), could warrant future research.

Although the empirical standard error (standard deviation of the point estimates) was required to calculate the relative error in the model-based standard errors, it was not presented separately. This was because if an estimator is biased (which was the case for the naive methods in many scenarios), it can affect the empirical standard error and complicate interpretation when compared to a reference method [4]. Future simulation studies may also consider using mean squared error and bias-eliminated coverage (in the presence of biased estimators). The former integrates both bias and empirical variance, penalising low variance for bias, and the latter helps understand the influence of bias on coverage [4]. For all performance measures except MISE, the accompanying MCSEs were calculated (see tables in Section E.2). These were not provided for MISE due to the computational burden and technical difficulty.

One limitation of the simulation study was the removal of (a very small number

of) problematic datasets. The observations were removed due to difficulties with the appropriate likelihood-based method for interval-censored data when there were only left- and right-censored events in both of the treatment groups (non-convergence) or only left-censored events in the treated group (high standard errors for the log hazard ratio). Alternate simulation designs could have been employed, for example, to ensure there were at least some interval-censored observations in both treatment groups. It should be noted that the appropriate likelihood-based method might not always be feasible when data are extreme.

A further limitation was that the motivating dataset investigated breast retraction opposed to the more common oncology endpoint of progression-free survival. This dataset was chosen due to its availability, although it is noted that the data generated in the simulation study may not be as generalisable to typical oncology trials as desired.

Previously, a lack of available software had hindered the use of appropriate methods for interval-censored data. Many options are now available including **stintreg** and **stintcox** in **Stata**, the latter recently becoming available with the release of **Stata** 17, and the **R** packages **icenReg** [134], **survival** [135], **icensBKL** [27] and **interval** [29]. There is therefore little reason to use naive imputation, given its inferior performance compared to an appropriate method for interval-censored data. Of the naive approaches, midpoint imputation frequently out-performed end imputation and was generally better than beginning imputation (although this varied with the scenarios). This is intuitive as midpoint imputation is the only naive approach that takes into account the length of the interval with the imputation. Despite this, it is often end imputation that is used in clinical trials when the date of the assessment is used as the progression date, which the simulation study has shown could result in biased estimates, artificially precise standard errors and confidence intervals that do not contain the true value.

### **3.8 Conclusion**

This chapter began by reviewing existing simulation studies to identify where knowledge was lacking in terms of the performance of naive imputation on interval-censored data. No study had investigated the survival probability at specified time points and this was investigated in a comprehensive simulation study. The simulation study found that, generally, midpoint imputation out-performed end imputation and often beginning imputation too (although this depended on the scenario). However, there were many scenarios where the naive methods performed poorly and an appropriate method for interval-censored data should be employed. A motivating dataset was analysed and corroborated the results of the simulation study. This chapter finished with a discussion, comparing the results of the simulation study to those identified in the literature review and highlighting the strengths, possible extensions and limitations of this work.

# Chapter 4

---

## Stabilised Versus Unstabilised Weights in an Inverse Probability Weighted Survival Analysis

---

### 4.1 Outline

This chapter illustrates the methods pertaining to IP weighting, as introduced in Section 2.6. The first objective is to compare the performance of stabilised and unstabilised weights in a motivating dataset. The second objective is to prove that the Kaplan-Meier estimator of marginal survival is equivalent when calculated with unstabilised and stabilised weights. The third objective is to confirm when stabilised and unstabilised weights give the same point estimates with a simulation study. If they differ, the simulation study will be used to confirm that both weights give unbiased estimates in a number of scenarios. The chapter finishes with a discussion and conclusion.

### 4.2 Introduction

IP weighting was introduced in Sections 1.1.3 and 2.6 in the context of survival data with a fixed, binary treatment/exposure. Examples of IP weighted analyses include investigating survival among women with cervical cancer exposed to a HIV infection [136] and survival among patients with colorectal liver metastases who underwent

laparoscopic versus open resection [137].

As discussed in Section 2.6.6, when performing an IP weighted analysis, two types of weights are frequently used to target the whole sampled population: stabilised and unstabilised. Point estimates of the marginal treatment/exposure effect obtained using stabilised and unstabilised weights are equivalent when the outcome model is saturated, for example, when the outcome and treatment/exposure are binary [15]. However, the same is not true when the outcome model employed is not saturated, for example, with marginal structural models (time-varying treatment/exposure and confounders) and continuous treatments/exposures, and in this case the two weights will result in different estimates [18]. Models for survival outcomes with a binary treatment/exposure can also be unsaturated. The difference in estimates from the two weights in this setting can be observed in Hajage *et al* [2] and will be demonstrated in Section 4.3. Henceforth, the term treatment effect will be used to represent a treatment or exposure effect.

In terms of point estimation, both stabilised and unstabilised weights in an IP weighted analysis provide unbiased estimates of the marginal treatment effect, given the assumptions in Section 2.6.3 hold [18]. Prior simulation studies have touched upon comparing point estimates from stabilised and unstabilised weights in the context of survival data. However, they investigated more complex settings, for example, marginal structural models [138, 139]. Xiao *et al* [139] found that both stabilised and unstabilised weights yielded practically unbiased estimates for the direct effect of current treatment while Westreich *et al* [138] concluded that stabilised weights gave essentially unbiased estimates, while unstabilised weights gave notably biased estimates when the treatment prevalence was less common (20%). In terms of an unsaturated IP weighted model for a survival outcome with a binary treatment, Hajage *et al*'s [2] simulation study demonstrated that point estimates were noticeably different between the weights when the treatment prevalence was small (10%), but this was not explored in detail.

Stabilised weights have, however, often been recommended in terms of variance estimation [15]. Hernan and Robins [15] recommend the use of stabilised weights as

they typically result in narrower 95% confidence intervals, that is, result in a smaller estimated variance. This is because unstabilised weights usually have more extreme weights, which cause the wider confidence intervals [18, 87]. However, this advice is given in the context of time-varying or continuous treatments and is not explicitly given in the context of unsaturated models for time to event outcomes and a fixed, binary treatment.

A comprehensive simulation study was performed by Austin [19] that investigated different variance estimators for stabilised and unstabilised weights in the context of a fixed, binary treatment with a survival outcome. The variance was underestimated when using the naive variance estimator with unstabilised weights, which can be explained by the variance being estimated on a pseudo-population twice as large as the original. However, stabilised and unstabilised weights gave similar variance estimates when either the robust or bootstrap variance estimator was used. This may suggest that in the setting of a survival outcome with a fixed, binary treatment (at least in the scenarios investigated in the simulation study) extreme weights may not be as prevalent as in marginal structural models, as stabilised weights did not appear to offer any benefit in terms of variance estimation.

In the setting of an unsaturated survival model with a fixed, binary treatment, there does not appear to be an advantage to using stabilised weights in terms of variance estimation, when an appropriate variance estimator is used, despite this being the main driver for the recommendation of stabilised weights in other settings. In addition, both weights provide unbiased estimators. However, noticeable differences in the point estimates from the weights have been observed, especially when the treatment prevalence is low. The first aim of the chapter is to perform a simulation study to confirm when stabilised and unstabilised weights give the same point estimates in the context of survival data (that is, to confirm when survival models are saturated). When stabilised and unstabilised weights give different estimates (that is, when survival models are unsaturated), the second aim is to confirm that neither weight gives a less biased estimate. To achieve these aims, the marginal estimands of interest first need to be chosen.

The marginal hazard ratio is commonly used to summarise the marginal treatment effect in survival data. As discussed in Section 2.6.2, the hazard ratio has attracted several criticisms as a treatment effect measure. Aalen *et al* [140] give further limitations of the Cox model from a causal point of view. The hazard ratio is also non-collapsible; meaning that, even in the absence of confounding, the conditional and marginal effects do not coincide [86].

One alternative is the difference in marginal RMST [40]. The difference in RMST has been advocated to quantify the treatment effect [40, 41, 141, 142], especially when the proportional hazards assumption is not valid. Cole and Hernan [143] and Xie and Liu [144] showed how an IP weighted Kaplan-Meier estimator can be used to provide marginal survival estimates. This can easily be extended to calculate the marginal RMST by integrating the marginal survival curve [145]. Alternatively, the marginal RMST can be calculated from a fitted marginal parametric model. Despite being an attractive alternative to the marginal hazard ratio in causal inference studies, the difference in marginal RMST is not yet widely used.

Most research has focused on the Cox model and the marginal hazard ratio. The performance of parametric models has rarely been evaluated and the extension to the difference in marginal RMST in any IP weighting simulation study is limited. As such, both the marginal log hazard ratio and difference in marginal RMST are investigated in this simulation study and various analysis approaches are used, including parametric models.

This chapter is structured as follows: Section 4.3 introduces the illustrative dataset and demonstrates how different point estimates can be produced when using stabilised and unstabilised weights. A proof regarding the equality of point estimates from the IP weighted Kaplan-Meier estimator for the two weights is given in Section 4.4. Section 4.5 describes the simulation study methods following the ADEMP [4] structure and Section 4.6 gives the simulation study results. The chapter finishes with a discussion and conclusion.

## 4.3 Illustrative Dataset

### 4.3.1 Description

The STD dataset was introduced in Section 1.3.5 and was conducted to investigate the risk factors associated with reinfection for patients with sexually transmitted diseases. The study consisted of 877 individuals and 16 covariates, along with the reinfection time and event indicator. The covariates were grouped into demographic, behavioural (recorded at the examination when the initial infection was diagnosed) and symptoms (noticed at the time of the initial infection) and were:

- **Demographic:** race, marital status, age at initial infection, years of schooling and type of initial infection.
- **Behavioural:** number of sexual partners in the last 30 days, oral sex within past 12 months, rectal sex within past 12 months and condom use.
- **Symptom:** abdominal pain, sign of discharge, sign of dysuria, sign of itch, sign of lesion, sign of rash and sign of lymph node involvement.

Similar to Xie and Liu [144], this analysis addresses the causal question of whether the time to reinfection differs by race (recorded as White (reference group) or Black). Note that this analysis is for demonstration purposes and is not a comprehensive analysis of the STD dataset.

### 4.3.2 Methods

The baseline data were summarised and standardised differences of the raw data were calculated. A logistic regression model was used to model race against the remaining 15 covariates (linear form only, no interactions). An improved analysis would incorporate expert knowledge and consider higher order terms and interactions. As mentioned in Section 2.6.5, alternatives to logistic regression could also have been considered. The improvements were not performed in this illustrative analysis, as the focus of this work was on the point estimates from using stabilised and unstabilised weights.

Once the logistic regression model was fitted, the unstabilised and stabilised weights were calculated following Equations 2.23 and 2.24. The standardised differences in the weighted dataset (for both weight types) were reviewed to investigate how well balance had been achieved. A standardised difference of 0.1 is often used as a guideline for the limit of acceptable imbalance [146, 147].

Two estimands were of interest: the marginal hazard ratio and difference in marginal RMST at 4 years. 4 years was chosen as it was close to the maximum follow-up time of 4.2 years. The following approaches were used for the estimands (unstabilised and stabilised weights were used for each approach):

- **Marginal hazard ratio:** IP weighted Cox and Weibull models, with race as the only covariate.
- **Difference in marginal RMST at 4 years:** IP weighted Weibull model and the IP weighted Kaplan-Meier estimator.

The proportional hazards assumption was checked using Schoenfeld residuals and by adding an interaction term with time into the IP weighted Cox model.

The focus of this chapter is on point estimation, variance estimation is discussed in detail in Chapter 5.

### 4.3.3 Results

Of the 877 individuals, 585 (66.7%) were Black and 292 (33.3%) were White. The baseline data and standardised differences for the raw data are given in Table 4.1. Number of sexual partners in the last 30 days was transformed into a categorical variable with  $\geq 3$  being grouped into a single level due to low frequency. Most covariates were reasonably balanced to moderately imbalanced. Three covariates had considerable imbalance: years of schooling, type of initial infection and oral sex within the past 12 months.

The unstabilised weights ranged from 1.02 to 14.36 with mean 1.98 and median 1.40. In comparison, the stabilised weights ranged from 0.35 to 4.78 with mean 0.99 and median 0.83. The standardised differences for the weighted dataset

**Table 4.1:** Baseline data and standardised differences of the raw and weighted data (using stabilised weights) in the STD dataset

Covariates	Value/ Statistic	Black	White	Standardised Difference	
		Individuals N = 585 (%)	Individuals N = 292 (%)	Raw	SW
Marital status	Married	16 (2.7)	12 (4.1)	0.099	0.086
	Single	532 (90.9)	257 (88.0)		
	Divorced/ Separated	37 (6.3)	23 (7.9)		
Age at initial infection	Mean (SD)	20.8 (5.7)	20.4 (4.8)	0.068	0.041
	Range	13, 46	13, 48		
Years of schooling	Mean (SD)	11.3 (1.6)	11.7 (1.8)	0.253	0.064
	Range	6, 17	6, 18		
Type of initial infection	Gonorrhea	97 (16.6)	43 (14.7)	0.556	0.232
	Chlamydia	215 (36.8)	181 (62.0)		
	Both	273 (46.7)	68 (23.3)		
Number of sexual partners in the last 30 days	0	52 (8.9)	18 (6.2)	0.122	0.063
	1	404 (69.1)	203 (69.5)		
	2	92 (15.7)	54 (18.5)		
	3+	37 (6.3)	17 (5.8)		
Oral sex within past 12 months <sup>1</sup>	Yes	120 (20.5)	168 (57.5)	0.819	0.298
Rectal sex within past 12 months <sup>1</sup>	Yes	25 (4.3)	26 (8.9)	0.187	0.102
Condom use <sup>1</sup>	Always	32 (5.5)	22 (7.5)	0.084	0.033
	Sometimes	343 (58.6)	168 (57.5)		
	Never	210 (35.9)	102 (34.9)		
Abdominal pain <sup>2</sup>	Yes	91 (15.6)	35 (12.0)	0.104	0.006
Sign of discharge <sup>2</sup>	Yes	276 (47.2)	129 (44.2)	0.06	0.032
Sign of dysuria <sup>2</sup>	Yes	67 (11.5)	47 (16.1)	0.135	0.004
Sign of itch <sup>2</sup>	Yes	103 (17.6)	60 (20.6)	0.075	0.063
Sign of lesion <sup>2</sup>	Yes	14 (2.4)	15 (5.1)	0.144	0.075
Sign of rash <sup>2</sup>	Yes	19 (3.3)	4 (1.4)	0.125	0.077
Sign of lymph involvement <sup>2</sup>	Yes	8 (1.4)	4 (1.4)	<0.001	0.032

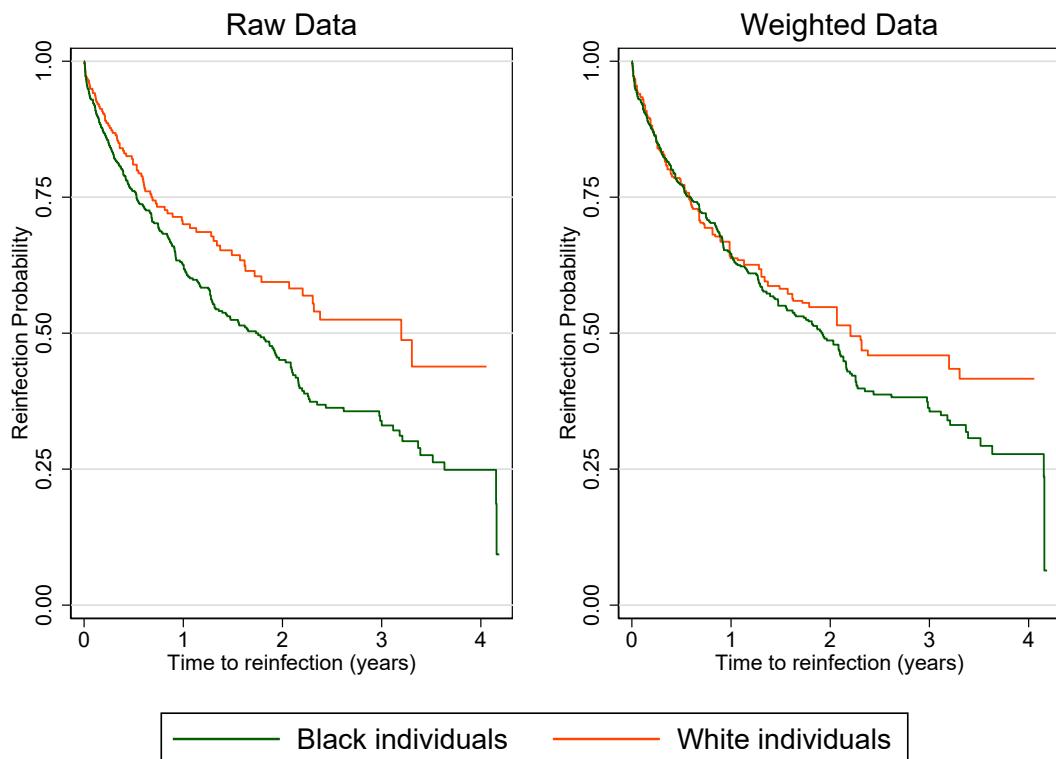
SW = Stabilised weights

<sup>1</sup>Recorded at the examination when the initial infection was diagnosed.

<sup>2</sup>Noticed at the time of the initial infection.

(using stabilised weights) are given in Table 4.1. Unstabilised weights produced extremely similar results. Covariates were more balanced in the weighted datasets, with most standardised differences below 0.1, except for type of initial infection, oral sex within past 12 months and rectal sex within past 12 months (borderline). Although the standardised differences for the former two were considerably higher than the proposed threshold, they were greatly reduced from the raw data.

The maximum follow-up time was 4.2 years and 347 (39.6%) of individuals became reinfected. The survival curves from the raw and weighted data are given in Figure 4.1. Unstabilised and stabilised weights gave identical IP weighted Kaplan-Meier estimates of the marginal survival curves and this is proved in the following section.



**Figure 4.1:** Kaplan-Meier curve of time to reinfection in the raw (left) and weighted (right) data by race for the STD data. Stabilised and unstabilised weights gave exactly the same weighted graph

The proportional hazards assumption was reasonable for the weighted datasets. The IP weighted Weibull model gave an estimated marginal hazard ratio of 1.1416

with unstabilised weights and 1.1403 with stabilised weights, while the IP weighted Cox model gave estimates of 1.1304 and 1.1276 for unstabilised and stabilised weights, respectively. All methods suggested that Black individuals had a 13-14% increased hazard of being reinfected.

The IP weighted Weibull model gave an estimated difference in marginal RMST at 4 years of -0.1626 and -0.1611 for the unstabilised and stabilised weights respectively. The IP weighted Kaplan-Meier estimate was -0.2161 for both weights. All methods suggested that Black individuals were reinfection free for 59-79 less days on average than White individuals.

**Table 4.2:** Estimates of the marginal hazard ratio and difference in marginal RMST at 4 years for the different methods and types of weights on the STD data

	Marginal Hazard Ratio		Difference in Marginal RMST at 4 Years	
	Unstabilised	Stabilised	Unstabilised	Stabilised
Weibull	1.1416	1.1403	-0.1626	-0.1611
Cox	1.1304	1.1276	-	-
KM	-	-	-0.2161	-0.2161

#### 4.3.4 Discussion

The analysis of the STD dataset has illustrated that unstabilised and stabilised weights give identical IP weighted Kaplan-Meier estimates of marginal survival, as this approach is saturated. This is proved in the following section and confirmed in the simulation study. The analysis showed that for unsaturated models, different point estimates can be obtained from the different weights. The simulation study in Sections 4.5 and 4.6 aims to confirm which survival models are saturated (and give identical results for the weights) and that both stabilised and unstabilised weights are unbiased in the setting of unsaturated survival models.

## 4.4 Equivalence of IP Weighted Kaplan-Meier Estimators

### 4.4.1 Standard Kaplan-Meier Estimator

Recall Equation 2.17 for the Kaplan-Meier estimate of the survival probability at time  $t$ , where  $d_j$  was the number of events occurring at time  $t_{(j)}$  and  $n_j$  was the number of individuals at risk at the time just before  $t_{(j)}$ :

$$\hat{S}(t) = \prod_{j:t_{(j)} \leq t} \left(1 - \frac{d_j}{n_j}\right)$$

Let  $Z$  denote the treatment variable. The Kaplan-Meier estimator for group  $k$  is (where  $\delta_i$  is the event indicator and  $1(\cdot)$  is the indicator function) [144]:

$$\begin{aligned} d_{jk} &= \sum_{i:t_i=t_{(j)}} \delta_i 1(Z_i = k) \\ n_{jk} &= \sum_{i:t_i \geq t_{(j)}} 1(Z_i = k) \\ \hat{S}_k(t) &= \prod_{j:t_{(j)} \leq t} \left(1 - \frac{d_{jk}}{n_{jk}}\right) \end{aligned}$$

### 4.4.2 Weighted Kaplan-Meier Estimator

The weighted Kaplan-Meier estimator for group  $k$  with weights  $w_{ik}$  becomes [144]:

$$\begin{aligned} d_{jk}^w &= \sum_{i:t_i=t_{(j)}} w_{ik} \delta_i 1(Z_i = k) \\ n_{jk}^w &= \sum_{i:t_i \geq t_{(j)}} w_{ik} 1(Z_i = k) \\ \hat{S}_k^w(t) &= \prod_{j:t_{(j)} \leq t} \left(1 - \frac{d_{jk}^w}{n_{jk}^w}\right) \end{aligned}$$

## Unstabilised weights

The unstabilised weights for treatment  $k$  are  $w_{ik} = 1/p_{ik}$  where  $p_{ik} = \text{P}(Z_i = k | \mathbf{x}_i)$  and  $\mathbf{x}_i$  is the observed vector of confounding values for individual  $i$  to be adjusted for. Note, that in the case of a binary treatment,  $p_{i1} = e_i$  and  $p_{i0} = 1 - e_i$ , where  $e_i$  is the propensity score as defined in Section 2.6.5. The IP weighted Kaplan-Meier estimator with unstabilised weights is [144]:

$$\begin{aligned} d_{jk}^u &= \sum_{i:t_i=t_{(j)}} \frac{1}{p_{ik}} \delta_i 1(Z_i = k) \\ n_{jk}^u &= \sum_{i:t_i \geq t_{(j)}} \frac{1}{p_{ik}} 1(Z_i = k) \\ \widehat{S}_k^u(t) &= \prod_{j:t_{(j)} \leq t} \left( 1 - \frac{d_{jk}^u}{n_{jk}^u} \right) \end{aligned}$$

## Stabilised weights

The stabilised weights for treatment  $k$  are  $w_{ik} = p_k / p_{ik}$  where  $p_k = \text{P}(Z = k)$ .  $p_k$  is the unconditional probability an individual is assigned to treatment  $k$ . It is proposed in this thesis that the IP weighted Kaplan-Meier estimator with stabilised weights is:

$$\begin{aligned} d_{jk}^s &= \sum_{i:t_i=t_{(j)}} \frac{p_k}{p_{ik}} \delta_i 1(Z_i = k) = p_k \sum_{i:t_i=t_{(j)}} \frac{1}{p_{ik}} \delta_i 1(Z_i = k) = p_k d_{jk}^u \\ n_{jk}^s &= \sum_{i:t_i \geq t_{(j)}} \frac{p_k}{p_{ik}} 1(Z_i = k) = p_k \sum_{i:t_i \geq t_{(j)}} \frac{1}{p_{ik}} 1(Z_i = k) = p_k n_{jk}^u \\ \widehat{S}_k^s(t) &= \prod_{j:t_{(j)} \leq t} \left( 1 - \frac{d_{jk}^s}{n_{jk}^s} \right) = \prod_{j:t_{(j)} \leq t} \left( 1 - \frac{p_k d_{jk}^u}{p_k n_{jk}^u} \right) = \widehat{S}_k^u(t) \end{aligned}$$

Hence, it has been proven in this thesis that stabilised and unstabilised weights give the same IP weighted Kaplan-Meier estimates for the marginal survival probability (and therefore marginal RMST) at time  $t$ . This aligns with the literature as the Kaplan-Meier approach is saturated.

## 4.5 Simulation Study Methods

### 4.5.1 Aims

1. To confirm that saturated models result in the same estimates for unstabilised and stabilised weights and to confirm which survival models are saturated in an IP weighted analysis with a time-to-event outcome.
2. To confirm that both stabilised and unstabilised weights are unbiased for unsaturated survival models with a fixed, binary treatment and to ensure this still holds in a range of scenarios.

### 4.5.2 Data Generating Mechanism

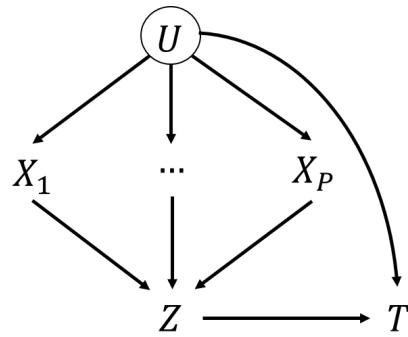
#### General Algorithm

Data generation followed Hajage *et al* 2018 [2], which in turn followed Hajage *et al* 2016 [148] and was inspired by Havercroft and Didelez [149]. The data generating mechanism simulates event times from a marginal outcome model. Following the discussion of marginal models in Section 2.6.7, this means that the event times are simulated from a model with treatment as the only covariate. The motivation for this approach is that the marginal hazard functions will be proportional and the true value of the marginal hazard ratio will be known. The corresponding analysis methods will, therefore, be correctly specified. There is confounding due to  $U$  being a common ancestor of treatment  $Z$  and outcome  $T$  [2], see the directed acyclic graph in Figure 4.2. This is adjusted for by conditioning on covariates  $X_1, \dots, X_P$ , as given the covariates, treatment  $Z$  is independent of  $U$  [149]. The outcome is then directly affected by treatment and indirectly affected by covariates  $X_1, \dots, X_P$  through their correlation with  $U$ .

The following modifications were made to the data generating mechanism in Hajage *et al* [2]:

- A Weibull model was used for the marginal outcome model, instead of the exponential model, in order to simulate more complex survival data.

- Following Austin [19], a bisection approach, instead of minimisation, was used to determine parameter  $\alpha_0$  so that the desired treatment prevalence was achieved.
- Censoring was applied by a combination of administrative censoring and drawing an intermittent censoring time from an exponential distribution. Hajage *et al* [2] only used intermittent censoring, drawn from a uniform distribution.



**Figure 4.2:** Directed acyclic graph representing the data generating mechanism, based on Figure 1 from Hajage *et al* [2]

The data generating mechanism is now described. First,  $n_{obs}$  patients were simulated.  $P+1$  normally distributed covariates  $X_1, \dots, X_P, X_U$  were then generated from the following multivariate normal distribution:

$$(X_1, \dots, X_P, X_U)^T \sim N(\mathbf{0}; \Sigma) \quad \Sigma = \begin{pmatrix} 1 & 0 & \dots & 0 & \sigma_1 \\ 0 & 1 & \dots & 0 & \sigma_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & \sigma_P \\ \sigma_1 & \sigma_2 & \dots & \sigma_P & 1 \end{pmatrix}$$

The covariates  $X_1, \dots, X_P$  were therefore independent of each other but were each correlated with the standard normal covariate  $X_U$  through parameters  $\sigma_p$ .

Covariate  $X_U$  was transformed into a uniformly distributed covariate  $U \sim \mathcal{U}(0, 1)$  by calculating the cumulative density function of  $X_U$ :  $U = F_{X_U}(u) = P(X_U < u)$ .  $U$  was still correlated with the covariates  $X_1, \dots, X_P$ .

Binary treatment variable  $Z$  was drawn from a Bernoulli distribution with prob-

ability  $p_Z$ ,  $Z \sim \mathcal{B}(p_Z)$ , where:

$$\text{logit}(p_Z) = \alpha_0 + \sum_{p=1}^P \alpha_p X_p$$

The event time  $T^*$  was assumed to follow a Weibull distribution with shape parameter  $\gamma$  and scale parameter  $\lambda$ .  $\beta$  is the marginal log hazard ratio. The event time was simulated from  $U$  as follows:

$$T^* = \left\{ \frac{-\log(U)}{\lambda \exp(\beta Z)} \right\}^{1/\gamma}$$

The intermittent censoring time  $C$  was drawn from an exponential distribution with rate parameter  $\lambda_C$ . Outcome  $T$  was set to  $T = \min(T^*, C, c_a)$ , where  $c_a$  was the administrative censoring time. The event indicator  $\delta$  was set as follows:

- $\delta = 1$  if  $T^* \leq C$  and  $T^* \leq c_a$
- $\delta = 0$  if  $T^* > C$  or  $T^* > c_a$

## Evaluated Scenarios

The following parameters were varied in the simulation study:

- Treatment prevalence:  $\pi_Z = \{0.1, 0.25, 0.5\}$
- Treatment effect:  $\exp(\beta) = \{0.5, 0.57, 0.67, 0.8, 1, 1.25, 1.5, 1.75, 2\}$
- No intermittent censoring and intermittent censoring rate parameter  $\lambda_C = 0.05$
- Sample size:  $n_{obs} = \{2000, 10000\}$

This led to a total of  $3 \times 9 \times 2 \times 2 = 108$  scenarios.

## Simulation Parameters

Where possible, parameters were chosen to follow Hajage *et al* [2]. It was set such that  $P = 9$  with parameters  $\alpha_p$  and  $\sigma_p$  taking values to represent different strengths

of association with the outcome and treatment [2]. The parameter values are given in Table 4.3. The parameters of the marginal Weibull outcome model were set to  $\gamma = 0.7$  and  $\lambda = 0.15$ . The administrative censoring time was set to  $c_a = 30$ .

**Table 4.3:** Parameters used in the data generating mechanism (treatment model and covariances), based on those from Hajage *et al* [2], Table 1

Parameter <sup>a</sup>	Value	Parameter <sup>b</sup>	Value
$\alpha_1$	$\log(1.25)$	$\sigma_1$	0.05
$\alpha_2$	$\log(1.50)$	$\sigma_2$	0.05
$\alpha_3$	$\log(1.75)$	$\sigma_3$	0.05
$\alpha_4$	$\log(1.25)$	$\sigma_4$	0.10
$\alpha_5$	$\log(1.50)$	$\sigma_5$	0.20
$\alpha_6$	$\log(1.75)$	$\sigma_6$	0.30
$\alpha_7$	$\log(1.10)$	$\sigma_7$	0.10
$\alpha_8$	$\log(1.10)$	$\sigma_8$	0.20
$\alpha_9$	$\log(1.10)$	$\sigma_9$	0.30

<sup>a</sup> $\alpha_p$ : coefficients of the treatment model.

<sup>b</sup> $\sigma_p$ : covariance between  $X_p$  covariates and  $X_U$ .

The parameter  $\alpha_0$  was specific to each desired treatment prevalence  $\pi_Z$  and was calculated using the bisection approach, as performed in Austin [19]. A sample of 10000 patients was simulated. The general data generating algorithm was performed up to and including the point treatment  $Z$  was allocated. An estimate  $\hat{\alpha}_0$  was then obtained using the bisection approach so that the desired treatment prevalence  $\pi_Z$  was achieved (using a tolerance of 0.000001). This process was repeated 1000 times and  $\alpha_0$  was set to the average of these estimates for each desired treatment prevalence  $\pi_Z$ . The resulting values were -2.59, -1.33 and 0 for treatment prevalences 10%, 25% and 50%, respectively.

### 4.5.3 Estimands

Two estimands were of interest:

1. The marginal log hazard ratio  $\beta$  (Equation 2.20 with the proportional hazards assumption)
2. The difference in the marginal RMST at time  $t = 20$ ,  $\Delta_\mu(20)$  (Equation 2.19)

The true value of the marginal log hazard ratio was simply the log hazard ratio  $\beta$  used in the data generating mechanism (as the data were generated from a marginal model).  $\gamma$  and  $\lambda$  used in the data generating mechanism were also the true shape and scale parameters respectively. The true marginal RMST in each group was calculated using numerical integration over 10000 equally spaced points in the interval  $[0, 20]$ . This resulted in a true marginal RMST in the control group of 10.29. The true marginal RMST in the treatment group, denoted  $\text{RMST}_1(t)$ , and therefore also the difference in marginal RMST, depended on the true marginal hazard ratio and is given in Table 4.4.

**Table 4.4:** True values of the marginal hazard ratio (HR), RMST in the treatment group at time 20 ( $\text{RMST}_1(20)$ ) and the difference in marginal RMST at time 20 ( $\Delta_\mu(20)$ ) in the data generating mechanism

Marginal HR, $e^\beta$	$\text{RMST}_1(20)$	$\Delta_\mu(20)$
0.50	14.15	3.86
0.57	13.51	3.22
0.67	12.66	2.37
0.80	11.65	1.36
1.00	10.29	0
1.25	8.87	-1.43
1.50	7.69	-2.60
1.75	6.71	-3.58
2.00	5.90	-4.39

#### 4.5.4 Methods

For all methods, the propensity score (as defined in Section 2.6.5) was estimated using logistic regression with all covariates  $X_1, \dots, X_9$  included. Unstabilised and stabilised weights were calculated using Equations 2.23 and 2.24, respectively. The following analysis approaches were then performed, depending on the estimand. In all cases, the outcome model had treatment  $Z$  as the only covariate.

1. IP weighted exponential model (Aim 1 only)
2. IP weighted Weibull model

3. IP weighted Cox model for the marginal log hazard ratio / IP weighted Kaplan-Meier estimator for the difference in marginal RMST

Each approach was used for both stabilised and unstabilised weights, resulting in 6 methods in total.

The exponential model was included as an analysis approach in this study to confirm whether it is saturated. As estimates were expected to be biased (the model was misspecified), the analysis approach was only used to address Aim 1.

RMST was calculated analytically for the exponential model and by numerical integration for the Weibull model.

#### 4.5.5 Performance Outcomes

The notation in Table 3.4 is used here. The following performance outcomes were analysed for each aim, where the true value of  $\theta$  was obtained as described in Section 4.5.3:

1. **Aim 1:** The mean of the point estimates  $E(\hat{\theta})$ .
2. **Aim 2:** The bias of the point estimates  $E(\hat{\theta}) - \theta$  and corresponding MCSE, as calculated in Equation 3.2.

#### Iteration Sample Size

An iteration sample size calculation was performed to ensure the bias (in Aim 2) was estimated to an acceptable degree of precision [4]. The iteration sample size can be calculated for a given value of MCSE of the bias using Equation 3.4. A maximum MCSE value of 0.005 for the marginal log hazard ratio and 0.02 for the difference in marginal RMST were deemed acceptable. The iteration sample size calculation was considered separately for the two sample sizes ( $n_{obs} = \{2000, 10000\}$ ), as it was expected that the  $n_{sim}$  needed for  $n_{obs} = 10000$  would be considerably smaller than for  $n_{obs} = 2000$  and would take consider longer to run for the same number of iterations.

For both sample sizes, an initial simulation with 1000 iterations was performed, in order to obtain estimates of  $\text{Var}(\hat{\theta})$ . Estimates of  $\text{Var}(\hat{\theta})$  were for each of the estimands, for each method (relevant to Aim 2) and for each of the 54 scenarios (within each sample size). The maximum over the scenarios and methods was taken for each estimand and the iteration sample size for the main study was chosen as the maximum across the estimands. The results are shown in Tables 4.5 and 4.6 for sample sizes 2000 and 10000, respectively.

**Table 4.5:**  $n_{obs} = 2000$ : Estimated maximum variance for each estimand from the preliminary simulation with 1000 iterations, corresponding  $n_{sim}$  required so that the MCSE for bias is less than or equal the acceptable value and the estimated maximum MCSE when  $n_{sim} = 2500$

Estimand	Maximum $\text{Var}(\hat{\theta})$	Acceptable MCSE	$n_{sim}$ needed	Maximum MCSE if $n_{sim} = 2500$
Marginal log HR	0.0275	0.005	1100	0.0033
<b>Difference in RMST</b>	<b>0.9746</b>	<b>0.02</b>	<b>2437</b>	<b>0.0197</b>

**Table 4.6:**  $n_{obs} = 10000$ : Estimated maximum variance for each estimand from the preliminary simulation with 1000 iterations, corresponding  $n_{sim}$  required so that the MCSE for bias is less than or equal the acceptable value and the estimated maximum MCSE when  $n_{sim} = 1000$

Estimand	Maximum $\text{Var}(\hat{\theta})$	Acceptable MCSE	$n_{sim}$ needed	Maximum MCSE if $n_{sim} = 1000$
Marginal log HR	0.0055	0.005	221	0.0024
Difference in RMST	0.1895	0.02	474	0.0138

The iteration sample size was determined to be 2500 for  $n_{obs} = 2000$ . This led to an estimated maximum MCSE of 0.0033 and 0.0197 for the marginal log hazard ratio and difference in marginal RMST across the scenarios/methods, respectively. The iteration sample size of 1000 was deemed sufficient for  $n_{obs} = 10000$ , as the maximum estimated MCSE for both estimands were already below the thresholds. This dataset was therefore used in the main analysis for the scenarios corresponding to  $n_{obs} = 10000$ .

#### 4.5.6 Software

All simulations were performed in **Stata/MP** 4-core, version 17. The survival data was set using **pw** weights. The IP weighted exponential model was fit using **streg**. The IP weighted Weibull model was fit using **stpm2** [42] with 1 degree of freedom (with the **noorthog** option specified), which is equivalent to a Weibull model programmed with **streg**. This command was chosen as RMST can easily be obtained with a postestimation command. The IP weighted Cox model was fit using **stcox** and the difference in marginal RMST was calculated from the IP weighted Kaplan-Meier estimator using **strmst2** [150].

The simulations were performed on the University of Leicester's High Performance Computing cluster, utilising 4 processor cores to improve efficiency. In total, three simulations were performed: a preliminary simulation used in the iteration sample size calculation for  $n_{obs} = 2000$ , the main simulation for  $n_{obs} = 2000$  and the main simulation for  $n_{obs} = 10000$ . Details are given in Table 4.7 along with the starting seeds and average computational time for each iteration. Note that the simulation is performed on all 54 scenarios within each iteration.

**Table 4.7:** Starting seeds and computational time for the preliminary and main simulation studies

Simulation	Sample size ( $n_{obs}$ )	Number of iterations ( $n_{sim}$ )	Starting seed	Average computational time per iteration(s)
Preliminary	2000	1000	4387562	48.9
Main	2000	2500	8264930	43.9
Main <sup>1</sup>	10000	1000	1048104	705.4

<sup>1</sup>This simulation was used for both the preliminary iteration sample size calculation and main analysis, see Section 4.5.5

## 4.6 Simulation Study Results

### 4.6.1 Exploratory Analysis

#### Errors and Convergence

All estimates from models converged. There were no errors for the scenarios where  $n_{obs} = 10000$ . Of the 270,000 difference in marginal RMST estimates calculated by the Kaplan-Meier approach in the  $n_{obs} = 2000$  scenarios (2500 repetitions  $\times$  54 scenarios  $\times$  2 types of weights), 196 produced an error ( $< 0.1\%$ ). This was due to there being no survival time (event or censored) greater than or equal to  $t = 20$  in at least one of the treatment groups and therefore the RMST in that group could not be calculated. This occurred in scenarios with the lowest treatment prevalence, three largest treatment effects and where there was intermittent censoring. Both the corresponding stabilised and unstabilised weights in these cases resulted in an error. The most prone scenario to error (treatment prevalence of 10%, hazard ratio of 2, intermittent censoring) produced an error in 75/2500 (3.0%) of the cases for each weight. As the percentage was small, these were excluded from the analysis. 105/108 (97.2%) scenarios (across both sample sizes) did not have an error. There were no other errors.

#### Monitoring Prevalences

The average actual treatment prevalences over the simulations for each scenario were consistent with the desired amount. The average amount of censoring over the simulations ranged between 13.3% to 31.1% across the scenarios when there was no intermittent censoring and between 34.6% to 53.7% when there was intermittent censoring. In the case of the latter, the average amount of intermittent censoring ranged from 31.6% to 46.8% and the average amount of administrative censoring ranged from 3.0% to 6.9%. All types of censoring increased with a smaller treatment effect and larger treatment prevalence.

## Weight Investigation

The average over the simulations of the maximum unstabilised weight ranged from 19.1 to 159.1 across the scenarios. Comparatively, the average of the maximum stabilised weight ranged from 9.6 to 15.9. The maximum unstabilised weight from any scenario and any simulation was 1172.2 compared to 209.4 for the stabilised weights. Scenarios with the higher sample size led to considerably larger average maximums of both weight types. Scenarios with a lower treatment prevalence had a noticeably larger average maximum unstabilised weight, while the average maximum stabilised weight was lower for scenarios with 10% prevalence compared to 25% for  $n_{obs} = 2000$  and similar for  $n_{obs} = 10000$ .

### 4.6.2 Main Analysis

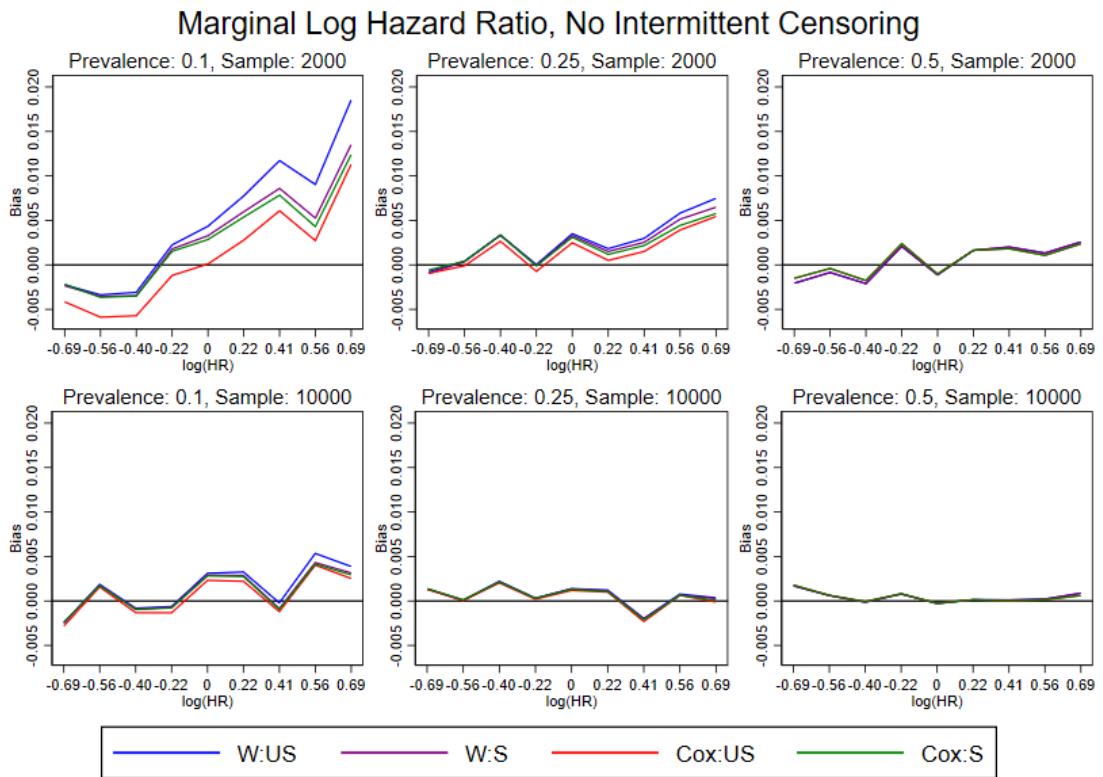
The maximum estimated MCSE for bias was 0.0034 and 0.0195 for the marginal log hazard ratio and difference in marginal RMST for  $n_{obs} = 2000$ , respectively. These were below the maximum acceptable MCSE thresholds of 0.005 and 0.02, as specified in Section 4.5.5, and suggests that the iteration sample size was sufficient. The average MCSE across the scenarios and methods (for those relevant to Aim 2) was 0.0019 and 0.0114 for the estimands, respectively.

#### Aim 1:

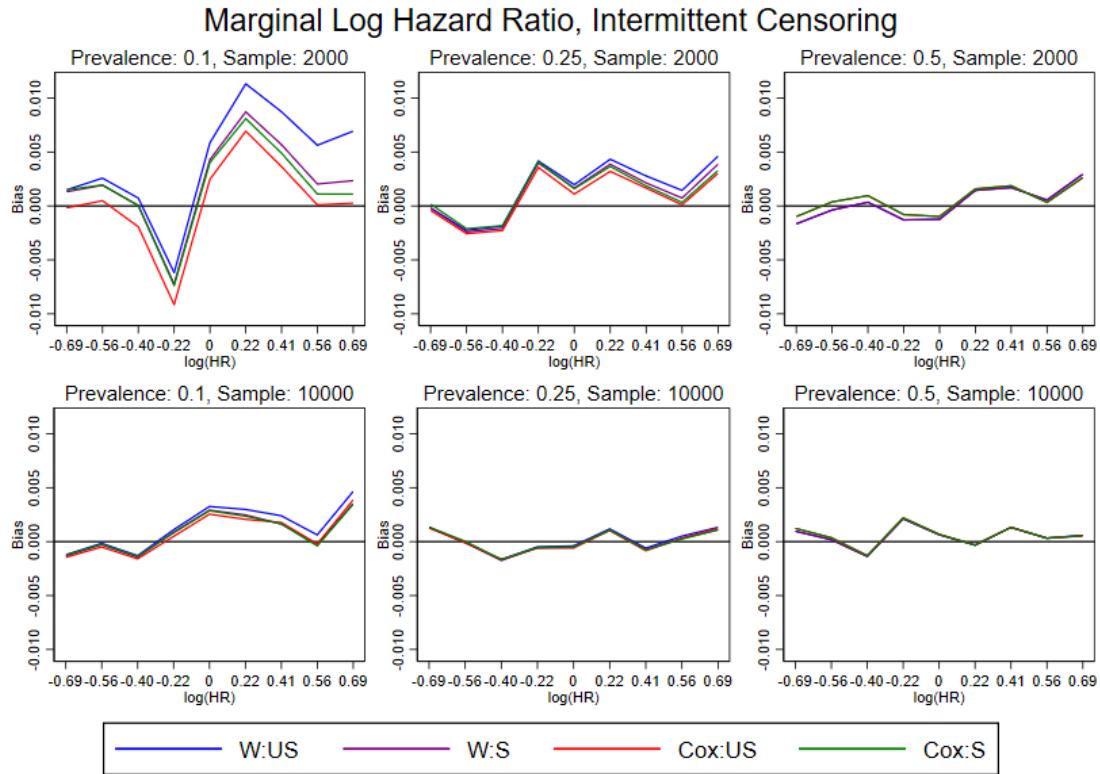
The IP weighted Kaplan-Meier estimator of the difference in marginal RMST gave identical point estimates from stabilised and unstabilised weights, confirming the proof in Section 4.4 and that this approach is saturated. The exponential model gave essentially the same estimates from the stabilised and unstabilised weights (the difference was  $< 0.00001$  for the marginal log hazard ratio and  $< 0.0001$  for the difference in marginal RMST for all simulations/scenarios), also confirming that this model is saturated with a fixed, binary treatment. The other methods gave different point estimates from stabilised and unstabilised weights, confirming that they are unsaturated, and this is demonstrated in Aim 2.

## Aim 2:

Figures 4.3 and 4.4 give the bias estimates for the marginal log hazard ratio for the scenarios with no intermittent censoring and those with intermittent censoring, respectively. Figures 4.5 and 4.6 give the corresponding results for the difference in marginal RMST. Tables F.1 and F.2 in the Appendix give the bias and MCSE for a subset of the scenarios for each estimand.

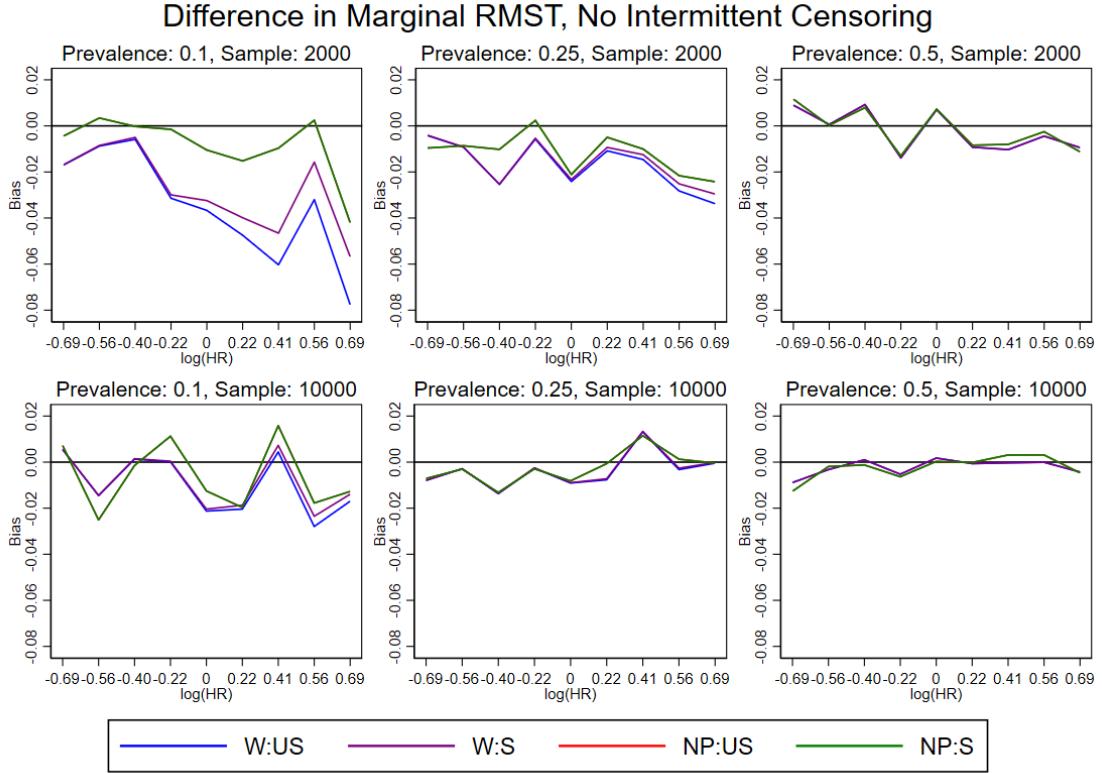


**Figure 4.3:** Simulation study results showing bias for the marginal log hazard ratio where censoring was administrative only. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, Cox:US Semi-parametric (Cox) unstabilised, Cox:S Semi-parametric (Cox) stabilised



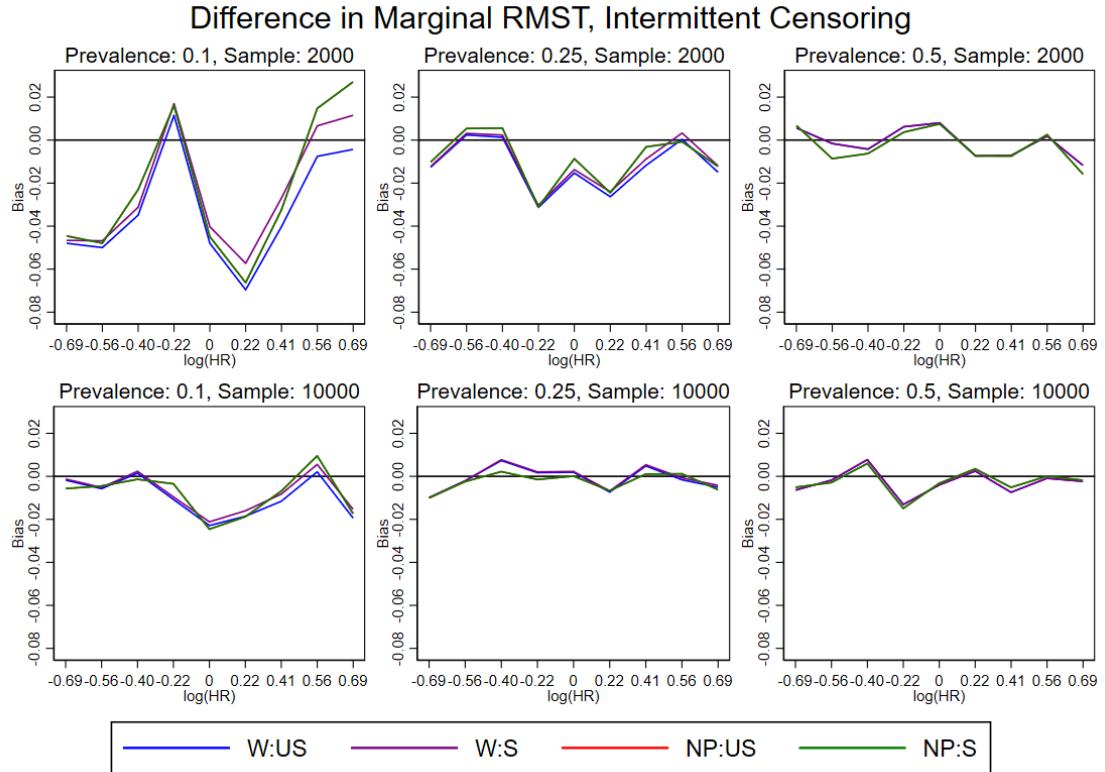
**Figure 4.4:** Simulation study results showing bias for the marginal log hazard ratio where censoring was both administrative and intermittent. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, Cox:US Semi-parametric (Cox) unstabilised, Cox:S Semi-parametric (Cox) stabilised

The absolute biases were small. In total, there were 432 estimates of bias ( $108 \text{ scenarios} \times 4 \text{ methods}$ ) for each estimand. 426 (98.6%) had an absolute bias of less than 0.01 for the marginal log hazard ratio. A maximum absolute bias of 0.01 (on the log scale) for a true hazard ratio of 0.5 would correspond to an estimated hazard ratio in the range (0.495, 0.505). For a true hazard ratio of 2, the range would be (1.980, 2.020). 408 (94.4%) scenarios had an absolute bias of less than 0.04 for the difference in marginal RMST. If the timescale is taken to be years, 0.04 would correspond to just over two weeks. Therefore, most estimates did not have a practically important magnitude of bias.



**Figure 4.5:** Simulation study results showing bias for the difference in marginal RMST where censoring was administrative only. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, NP:US Non-parametric (Kaplan-Meier) unstabilised, NP:S Non-parametric (Kaplan-Meier) stabilised. The red line (Kaplan-Meier estimator, unstabilised weights) cannot be seen as it is completely overlaid by the corresponding stabilised estimates.

The figures do, however, demonstrate some trends in bias across the scenarios. For both estimands, a smaller treatment prevalence and smaller sample size resulted in more biased results. In the scenarios where there was no intermittent censoring, a sample size of 2000 and a treatment prevalence of 10%, absolute bias increased as the treatment effect increased. This trend was observed to a lesser extent in the corresponding scenarios with a treatment prevalence of 25%. Generally, the censoring scheme did not appear to have any further impact on the bias.



**Figure 4.6:** Simulation study results showing bias for the difference in marginal RMST where censoring was both administrative and intermittent. From left to right, the treatment prevalence in the panels is 10%, 25% and 50%. The top and bottom panels show results for sample sizes 2000 and 10000, respectively. Within each panel, the log hazard ratio (treatment effect) is varied. W:US Weibull unstabilised, W:S Weibull stabilised, NP:US Non-parametric (Kaplan-Meier) unstabilised, NP:S Non-parametric (Kaplan-Meier) stabilised. The red line (Kaplan-Meier estimator, unstabilised weights) cannot be seen as it is completely overlaid by the corresponding stabilised estimates

In most scenarios, stabilised and unstabilised weights, from the unsaturated methods, gave similarly biased results. The differences between the weights were greater in the scenarios with a smaller treatment prevalence, smaller sample size and larger treatment effect. Generally, stabilised weights with the IP weighted Weibull model offered slightly less bias than the corresponding unstabilised weights, although this was only of practical importance when the treatment prevalence was 10%, sample size was 2000 and the marginal log hazard ratio was 0 or greater. In this case, however, the reverse was observed for the IP weighted Cox model for the marginal log hazard ratio. It is likely that the small biases observed can be considered ‘small sample biases’ and it is expected that they would go to zero if the

sample size were increased, with all other parameters remaining the same.

## 4.7 Discussion

This study found that both stabilised and unstabilised weights gave generally unbiased estimates as part of an IP weighted analysis on survival data with a fixed, binary treatment, confirming what was expected from the literature. In the majority of scenarios investigated in the simulation study, stabilised and unstabilised weights performed similarly well. The IP weighted Kaplan-Meier estimator for marginal survival and RMST gave identical results for stabilised and unstabilised weights, as proved in Section 4.4 and confirming that this approach is saturated, and the IP weighted exponential model gave near identical results, also confirming that this is a saturated model. In some scenarios, for both estimands, the IP weighted Weibull model with stabilised weights gave very slightly less biased results than the corresponding unstabilised weights. The only scenarios where this may be practically important were for a treatment prevalence of 10%, a sample size of 2000 and for a marginal log hazard ratio of greater than 0. This was consistent with what was seen in the marginal structural model setting, where unstabilised weights gave notably biased estimates when the treatment prevalence was less common [138]. Neither weight performed consistently better than the other for the IP weighted Cox model, which could also be concluded from Figure 2 and Appendix Figure 1 in Hajage *et al* [2].

An additional, useful metric from simulation studies is the empirical standard error, which is the standard deviation of the point estimates. The empirical standard error was estimated for each estimand, method and scenario. This resulted in 2 estimands  $\times$  3 methods (for each weight)  $\times$  108 scenarios = 648 empirical standard errors for each weight. For the unstabilised weights, the minimum and maximum empirical standard error were 0.0223 and 1.0718, with an mean of 0.2464, while the minimum and maximum were the same for stabilised weights, with an average of 0.2460. In addition to both weights providing generally unbiased estimates across

the scenarios, there did not appear to be an advantage with either weight in terms of a lower empirical standard error.

Despite generally low values of absolute bias, some trends were identified, which was unexpected as the models were correctly specified (the data were generated from a marginal proportional hazards model). Bias appeared to be greater for smaller prevalences, smaller sample sizes and greater treatment effects. Scenarios with a treatment prevalence of 10%, sample size of 2000, marginal log hazard ratio of 0 or greater and when there was no intermittent censoring had the highest absolute bias for both estimands. Hajage *et al* [2] also found these trends (Figure 2 and Appendix Figure 1). The authors reported that bias was less than 0.005 for all simulated scenarios, which was also true for the 10000 sample size scenarios in this study when the IP weighted Cox model was used to estimate the marginal log hazard ratio. Hajage *et al* [148] investigated the impact of rare exposure on IP weighting and also found that bias decreased as prevalence and sample size were increased. These findings are therefore consistent with published work and suggest caution is needed when using IP weighting with rare exposures and small sample sizes.

This work builds on the simulation study of Hajage *et al* [2] by considering the difference in marginal RMST as an estimand and by including the analysis of parametric survival models. Interest in RMST is growing, for example, in simulation studies comparing causal inference methods [151] and also in methodology development [145]. As well as being a single summary of the absolute risk, the difference in marginal RMST has the benefit of being a collapsible estimand.

In the simulation study, data were generated from a marginal outcome model, where the covariates  $X_1, \dots, X_P$  indirectly affected the outcome  $T$  through their correlation with  $U$ . This meant that the marginal models used to analyse the data were correctly specified. The true value of the estimands were easy to obtain and did not require estimating from a very large dataset.

Although generating from a marginal outcome model has advantages, the method described here has been criticised due to the lack of direct effect from covariates to treatment. It does not follow the natural sequential data generating mechanism,

where first  $U$  is generated, then  $X_1, \dots, X_P$ , then treatment  $Z$  and then outcome  $T$  [152]. These sequential mechanisms generate data from a conditional proportional hazards model and the resulting marginal model is of a more complex form; Keogh *et al* [152] shows the resulting marginal model when the conditional Cox model is used to generate data. Keogh *et al* [152] instead advocated the use of additive hazard models as their coefficients are collapsible. In this study, the more widely used proportional hazards class of models was chosen, although the restrictive setting in which the data were generated is acknowledged. A further limitation is that the confounders  $X_1, \dots, X_P$  were generated to be uncorrelated.

As with any simulation study, further scenarios could have been explored. For example, the amount of censoring could have been varied further, the parameters of the marginal survival model could have been varied and/or event times could have been generated from a more complex distribution, for example, Royston-Parmar models. Similarly, the associations of the variables  $X_p$  with the outcome,  $\sigma_p$ , and the treatment,  $\alpha_p$ , could have been varied. Although an uncommon prevalence of 10% was investigated, rare treatment or exposure prevalences could have been investigated, similar to the work of Hajage *et al* [148]. For example, exposure prevalence in large studies using electronic health records may be as small as 0.1%. Finally, the difference in marginal RMST was calculated at just one time point - other time points could have been explored.

In the complex setting of marginal structural models, stabilised weights are advocated, due to them producing less extreme weights, less biased estimates (in some scenarios) and smaller variances. In the simpler case, where treatment is fixed at baseline, it was expected that both stabilised and unstabilised weights would provide unbiased point estimates. This work has confirmed that in most scenarios either stabilised or unstabilised weights can be used to obtain point estimates when the outcome is survival. If the sample size and treatment prevalence are small, IP weighting should be used with care and possibly estimates from both stabilised and unstabilised weights calculated and compared.

## 4.8 Conclusion

This chapter began by demonstrating that IP weighting with unstabilised and stabilised weights can lead to slightly different point estimates when applied to survival data with a fixed, binary treatment. The IP weighted Kaplan-Meier estimator for marginal survival probabilities gives identical results for the different weights, as proved in Section 4.4 and confirmed in the simulation study. The simulation study was performed to confirm that both unstabilised and stabilised weights give unbiased estimates. In most scenarios, stabilised and unstabilised weights gave very similar and unbiased results, with caution needed in scenarios where the sample size and treatment prevalence were low. This chapter focused on point estimation. Variance estimation is explored in detail in the next chapter, with a new variance estimator proposed and the illustrative dataset and simulation study in this chapter extended.

# Chapter 5

---

## Closed-form Variance Estimator for Inverse Probability Weighted Parametric Survival Models

---

### 5.1 Outline

Chapter 4 investigated point estimation in IP weighted survival models. This chapter focuses on the corresponding variance estimation and begins by reviewing existing variance estimators. The first objective of the chapter is to propose a closed-form variance estimator for a range of IP weighted parametric survival models, which uses M-estimation to account for the associated uncertainty in the weight estimation. The second objective is to evaluate the performance of the proposed variance estimator, compared to the robust and bootstrap variance estimators, by extending the simulation study in Chapter 4. The final objective is to illustrate the proposed variance estimator on three datasets. The chapter finishes with a discussion and possible extensions.

### 5.2 Introduction

Chapter 4 investigated point estimation in IP weighted survival models. This chapter focuses on the corresponding variance estimation. Correct variance estimation is necessary for valid inference. An IP weighted analysis is a two-stage procedure,

where first the weights are estimated and then the weighted data are analysed. For successful variance estimation, the associated uncertainty from the first stage (weight estimation) should be incorporated into the analysis of the second stage. Historically, robust standard errors were proposed to address the within-subject correlation induced by weighting [153]. However, this estimator is conservative [15], as confirmed by Austin [19] in a comprehensive simulation study, and does not incorporate the associated uncertainty from the first stage. Instead, Austin recommended the bootstrap variance estimator, as it accurately estimated the sampling variability (of the log-hazard ratio) and generally resulted in confidence intervals with approximately correct coverage rates [19].

Austin noted that a closed-form variance estimator for IP weighted survival models, which appropriately takes into account the associated uncertainty in the weight estimation (first stage), would be useful. As Hajage *et al* notes [2], closed-form variance estimators can be an attractive alternative to bootstrapping, especially when computational time or reproducibility are important. Appropriate closed-form treatment effect variance estimators have already been proposed utilising M-estimation by Lunceford and Davidian [154] for continuous outcomes and Williamson *et al* [82] for binary outcomes. For survival outcomes, Hajage *et al* [2] proposed an appropriate closed-form variance estimator for the marginal hazard ratio by linearising the Cox model. Shu *et al* [21] proposed an asymptotically equivalent estimator based on M-estimation. Finally, Mao *et al* [20] combined Poissonisation of the Cox model (using penalised splines) with M-estimation to estimate a range of marginal survival estimands.

The recent developments of Hajage *et al* [2] and Shu *et al* [21] are specific to the Cox model and marginal hazard ratio. Hazard ratios have received criticism, as mentioned in Sections 2.6.2 and 4.2, particularly when used in a causal framework. Alternatives, such as the difference in marginal survival probabilities and RMST, have been advocated to quantify the treatment effect [40, 41, 141, 142], as mentioned in Section 4.2.

Non-parametric and parametric methods can more easily provide estimates of

the marginal survival function and RMST than the Cox model. In terms of non-parametric estimation, an IP weighted Kaplan-Meier estimator of marginal survival can be used [143, 144]. This was extended by Connor *et al* [145] to provide an estimate of marginal RMST by integrating the IP weighted Kaplan-Meier estimates. However, the corresponding variance estimator did not take into account the associated uncertainty in the weight estimation. In terms of parametric estimation, Mao *et al* [20] used penalised splines to Poissonise the Cox model and obtain estimates of the marginal survival function and RMST. Their variance estimator appropriately accounted for the associated uncertainty in the weight estimation; however, was only provided for the aforementioned model and not for a general framework of parametric survival models.

This work extends the work of Mao *et al* [20] to a range of IP weighted parametric models, instead of approximating the Cox model. Along with standard proportional hazards survival models (exponential, Weibull, Gompertz), AFT models (log-normal and log-logistic) can be accommodated in this framework, as well as flexible Royston-Parmar models. M-estimation is used to estimate the variance of all parameters in the weighted survival model and, like Mao *et al* [20], the variance of a suite of marginal estimands can subsequently be obtained using the delta method.

This chapter is structured as follows: Section 5.3 describes the current methods of variance estimation in an IP weighted survival model. Section 5.4 introduces M-estimation, proposes the closed-form variance estimator for IP weighted parametric survival models and gives a worked example for an IP weighted Royston-Parmar model. Sections 5.5 and 5.6 evaluate the performance of the proposed variance estimator in a simulation study. Section 5.7 illustrates the proposed variance estimator with three applications. The chapter finishes with a discussion, detailing areas for future work and a conclusion. A new **Stata** command, **stipw**, has been written to calculate the proposed variance estimator and this is discussed in Chapter 6.

## 5.3 Review of Current Methods

### 5.3.1 Naive Variance Estimator

The naive model-based variance estimator does not take into account the use of weights (and potential lack of independence in the pseudo-population [153]), nor the associated uncertainty in the weight estimation. The naive variance estimator, when unstabilised weights are used, can result in artificially precise variance estimates, as demonstrated in Austin's simulation study [19]. This is because the estimates are based on a pseudo-population twice the size of the original population. Alternatively, when stabilised weights are used, the pseudo-population is the same size as the original population. Although a marked improvement to unstabilised weights, the naive variance estimator with stabilised weights still did not have the desired properties in the simulation study (average model-based standard error equal to the empirical standard error and coverage of 95%) [19]. The most common approaches for variance estimation are therefore the robust and bootstrap variance estimators.

### 5.3.2 Robust Variance Estimator

Robust standard errors were first proposed independently by Huber [155] and White [156, 157] and were extended for use with the Cox model by Lin and Wei [158]. They are referred to as sandwich estimators (because of how the calculation appears) or robust estimators (as they are robust to model misspecification) [63]. They are robust to model misspecification in the sense that the model need not be true and the data need not be independent and identically distributed (i.i.d.). However, careful interpretation of the model parameters is needed when using robust standard errors [63].

In the case of IP weighting, robust standard errors are used to address the within-subject correlation induced by weighting [153], that is, to address the lack of independence in the pseudo-population due to the replication of individuals [19]. However, it has often been noted that robust standard errors are conservative. Austin's

simulation study demonstrated that robust standard errors can often result in wide confidence intervals that cover the true value more than 95% of the time [19]. They also do not take into account the associated uncertainty in the weight estimation.

### 5.3.3 Bootstrap Variance Estimator

Bootstrapping can be used to estimate the variance of the estimator  $\widehat{\theta}$ . This is a popular approach when methods based on exact estimates or large-sample approximations are too complicated to be calculated, Chapter 13 [15]. The bootstrap algorithm for an IP weighted analysis is as follows for a dataset of size  $n$ :

1. Sample  $n$  patients from the dataset with replacement. This gives sample  $b_i$ , the  $i^{th}$  bootstrap sample.
2. Perform the IP weighted analysis on the sample  $b_i$  to obtain the marginal estimand of interest  $\widehat{\theta}_i$ .
3. Repeat steps 1 and 2  $m$  times.
4. Calculate  $\widehat{se}(\widehat{\theta})$ , which is the standard deviation of the bootstrap estimates  $\widehat{\theta}_i$   
 $i = 1, .., m$ .

Bootstrapping can be computationally intensive, especially if  $n$  is very large. Robins and Hernan, Chapter 13 [15], use  $m = 1000$  bootstrap samples in an example but note that most publications only use 200-500 samples for the reason just given. For example, 200 [2, 19, 20] or 500 [21] bootstrap samples were used in recent simulation studies for IP weighted analyses on survival data.

### 5.3.4 Closed-form Variance Estimators

Three closed-form alternatives to bootstrapping, which appropriately take into account the associated uncertainty in the weight estimation, have been proposed for survival outcomes. Both Hajage *et al* [2] and Shu *et al* [21] have proposed variance estimators for the marginal hazard ratio from an IP weighted Cox model. Hajage *et*

*al*'s estimator was obtained using the influence function linearisation technique developed by Deville [159]. This involved linearising both the Cox model and propensity score weights. Shu *et al* [21] provided an asymptotically equivalent estimator based on estimating equations. M-estimation is more challenging for the Cox model as the partial likelihood score function is not a sum of i.i.d. terms [21]. The authors overcame this issue by adapting the strategy of Lin and Wei [158] and Binder [160]. They also extended their variance estimator to handle clustered data.

Mao *et al* [20] proposed to Poissonise the Cox model and utilise M-estimation to derive variance estimators for a range of marginal estimands. Their approach was based on the penalised spline estimation of the survival distribution [161], which requires the specification of the number and location of the knots. The marginal survival function is estimated for each treatment group separately and therefore does not require the proportional hazards assumption. The variance for the related estimands, such as the marginal RMST, can be estimated using the delta method. An alternate set of estimating equations is used to estimate the marginal hazard ratio, as this does require a single model to be fitted with treatment as a covariate.

## 5.4 M-estimation Variance Estimator for IP Weighted Parametric Survival Models

### 5.4.1 Notation, Estimands and Assumptions

Recall the notation proposed in Section 2.6.1. Following Williamson *et al* [82], consider a dataset with  $n$  individuals. Let  $Z$  denote the binary treatment allocation and let  $X = \{X_1, \dots, X_p\}$  be the set of  $p$  measured baseline confounders. Let the survival time  $T = \min(T^*, C)$  be the minimum of the event time  $T^*$  and censoring time  $C$  and let  $\delta$  be the event indicator. The standard assumption of independent censoring conditional on treatment variable  $Z$  is made.

Let  $Z_i$ ,  $T_i$  and  $\delta_i$  be the corresponding variables for individual  $i$ . Let  $\mathbf{X}_i = (X_{i0}, X_{i1}, \dots, X_{ip})$  be the vector of confounder variables for individual  $i$ , where  $X_{i0} =$

1 for the intercept term.

Let  $z_i$  be the observed treatment value for individual  $i$  and let  $\mathbf{z}$  be the vector of observed treatment values for all  $n$  individuals. Let  $t_i$  be the observed survival time and let  $\delta_i$  also represent the observed event indicator for individual  $i$ , where appropriate. Let  $\mathbf{X}$  be the  $n \times (p+1)$  matrix of confounder values with an additional column of ones for the intercept term and let  $\mathbf{x}_i = (x_{i0}, x_{i1}, \dots, x_{ip})$  be the vector of observed confounder values (with  $x_{i0} = 1$  for the intercept) for individual  $i$ .

Let the counterfactual outcomes for  $T$  be denoted by  $T^1$  and  $T^0$ . In causal inference, interest lies in contrasts between the counterfactual outcomes. In Section 2.6.2, a number of contrasts in marginal estimands for survival analysis were proposed: the difference in marginal survival probabilities (Equation 2.18), difference in marginal RMST (Equation 2.19) and the marginal hazard ratio (Equation 2.20 with the proportional hazards assumption).

The standard assumptions necessary for IP weighting are made: consistency, positivity and conditional exchangeability. It is also assumed that there is no model misspecification in the treatment model and outcome model. These are described in Section 2.6.3. In addition, the data  $(T_i, \delta_i, Z_i, \mathbf{X}_i)$  are assumed to be independently distributed and the stable unit treatment value assumption (SUTVA) is made.

### 5.4.2 Estimation of IP Weighted Parametric Survival Models

Sections 2.6.4-2.6.7 describe the algorithm for IP weighted parametric survival models. In short: the treatment variable  $Z$  is modelled incorporating the measured confounders  $X$  using logistic regression to estimate the propensity score (Equation 2.21). This involves estimating the logistic regression parameters  $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \dots, \alpha_p)$ . If stabilised weights will be used, the (unconditional) probability of treatment  $\pi = P(Z = 1)$  is required. This involves estimating  $\alpha_S$  (Equation 2.22), which is the intercept in a logistic model for treatment with no covariates. Either unstabilised (Equation 2.23) or stabilised (Equation 2.24) weights are then calculated

using the estimated propensity score.

Finally, a weighted survival model is used to model survival outcome  $T$  with treatment  $Z$  as the only covariate. The parameters  $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_q)$  for the weighted survival model can be estimated by maximising the weighted log-likelihood (Equation 2.25), weighted with unstabilised or stabilised weights. Note that the  $\boldsymbol{\beta}$  parameters are all the parameters in the outcome parametric model, not just covariate coefficients. At least one parameter in  $\boldsymbol{\beta}$  corresponds to the treatment effect (for example,  $\beta_1$ ). The remaining parameters correspond to the distributional parameters. Examples of what the  $\boldsymbol{\beta}$  correspond to for standard parametric models are given in Appendix G.1 and for Royston-Parmar models in Section 5.4.5.

### 5.4.3 M-estimation Framework for IP Weighted Parametric Survival Models

M-estimation is a versatile tool used to estimate the variance of a vector of parameters  $\boldsymbol{\theta}$  and a useful tutorial has been published by Stefanski and Boos [162]. In the context of IP weighting for parametric survival models,  $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$  for unstabilised weights or  $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \alpha_s, \boldsymbol{\beta})$  for stabilised weights.

In order to utilise M-estimation, a set of estimating equations needs to be defined. The process summarised in Section 5.4.2 and described in Sections 2.6.5, 2.6.6 and 2.6.7 is equivalent to simultaneously solving the estimating equations below for the parameters  $\boldsymbol{\theta}$ :

$$\sum_{i=1}^n \mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \mathbf{0}$$

For unstabilised weights,  $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$ . The notation for the propensity score  $e_i$  for individual  $i$  has been updated to  $e_i(\mathbf{X}_i, \boldsymbol{\alpha})$  to highlight the dependence of the propensity score on the  $\boldsymbol{\alpha}$  parameters. The estimating equations using unstabilised

weights are:

$$\mathbf{u}(\boldsymbol{\alpha}, \boldsymbol{\beta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ \frac{\partial l_i^u(\boldsymbol{\beta})}{\partial \beta_1} \\ \vdots \\ \frac{\partial l_i^u(\boldsymbol{\beta})}{\partial \beta_q} \end{pmatrix} \quad (5.1)$$

The first row in Equation 5.1 corresponds to the contribution of individual  $i$  to the score function of the treatment model. That is, the contribution of individual  $i$  to the derivative of the log-likelihood for a logistic regression model for treatment incorporating the confounders. The derivative is taken with respect to each treatment model parameter  $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \dots, \alpha_p)$  and will therefore be of length  $p + 1$  rows. The remaining rows in Equation 5.1 correspond to the contribution of individual  $i$  to the score function of the weighted outcome survival model. That is, the contribution of individual  $i$  to the derivative of Equation 2.25 with respect to each outcome model parameter  $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_q)$ .

For stabilised weights,  $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \alpha_s, \boldsymbol{\beta})$ . The notation for the unconditional probability of treatment  $\pi$  has been updated to  $\pi(\alpha_s)$  to highlight that the probability of treatment is a function of  $\alpha_s$ . The estimating equations using stabilised weights are:

$$\mathbf{u}(\boldsymbol{\alpha}, \alpha_s, \boldsymbol{\beta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ Z_i - \pi(\alpha_s) \\ \frac{\partial l_i^s(\boldsymbol{\beta})}{\partial \beta_1} \\ \vdots \\ \frac{\partial l_i^s(\boldsymbol{\beta})}{\partial \beta_q} \end{pmatrix} \quad (5.2)$$

The second row in Equation 5.2 corresponds to the contribution of individual  $i$  to the score function of a logistic model for treatment with no covariates. The derivative of the log-likelihood is with respect to parameter  $\alpha_s$ .

The solution to the estimating equations,  $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{\alpha}}, \widehat{\boldsymbol{\beta}})$  (unstabilised) or  $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{\alpha}}, \widehat{\alpha}_s \widehat{\boldsymbol{\beta}})$  (stabilised), are termed M-estimators. For IP weighted survival models, utilising M-estimation will incorporate the associated uncertainty in the weight estimation (that is, the associated uncertainty in the estimation of  $\boldsymbol{\alpha}$  and, for stabilised

weights,  $\alpha_s$ ) in the variance of the weighted survival model parameter estimates  $\hat{\beta}$ .

#### 5.4.4 General M-estimation Variance Estimator for IP Weighted Parametric Survival Models

The variance of the M-estimators  $\hat{\theta}$  can be estimated using the following equation [82, 162]:

$$\begin{aligned}\hat{\mathbf{A}} &= \frac{1}{n} \sum_{i=1}^n -\frac{\partial \mathbf{u}_i}{\partial \theta} \Big|_{\hat{\theta}} \\ \hat{\mathbf{B}} &= \frac{1}{n} \sum_{i=1}^n \mathbf{u}_i(\hat{\theta}) \mathbf{u}_i(\hat{\theta})^T \\ \hat{\mathbf{V}}(\hat{\theta}) &= \frac{1}{n} \hat{\mathbf{A}}^{-1} \hat{\mathbf{B}} \hat{\mathbf{A}}^{-T}\end{aligned}\tag{5.3}$$

##### Unstabilised weights

For IP weighted survival models with unstabilised weights,  $\mathbf{A}$  is a  $(p+q+1) \times (p+q+1)$  matrix. It can be split into four quadrants where  $\mathbf{0}$  is a  $(p+1) \times q$  matrix of zeros:

$$\begin{pmatrix} \mathbf{A}_1 & \mathbf{0} \\ \mathbf{A}_2 & \mathbf{A}_3 \end{pmatrix}$$

$\mathbf{A}_1$  is a  $(p+1) \times (p+1)$  symmetrical matrix. It corresponds to the (negative of the) treatment model estimating equation (first row in Equation 5.1) averaged over all individuals being differentiated with respect to the logistic regression coefficients  $\alpha$ . This can be estimated as follows, where  $\odot$  represents elementwise multiplication:

$$\mathbf{A}_1 = \frac{1}{n} \mathbf{X}^T \frac{\mathbf{X} \odot \exp(\mathbf{X} \hat{\alpha}^T)}{\{1 + \exp(\mathbf{X} \hat{\alpha}^T)\}^2}\tag{5.4}$$

$\mathbf{A}_2$  is a  $q \times (p+1)$  matrix. It corresponds to the (negative of the) outcome model estimating equations (second row onwards in Equation 5.1) averaged over all individuals being differentiated with respect to the logistic regression coefficients  $\alpha$ .

This can be estimated as follows:

$$\mathbf{A}_2 = \frac{-1}{n} \left\{ \frac{\partial \mathbf{l}^u(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\theta}}} \right\}^T \mathbf{X} \odot [\exp(\mathbf{X}\hat{\boldsymbol{\alpha}}^T) - \mathbf{z} \odot \{\exp(-\mathbf{X}\hat{\boldsymbol{\alpha}}^T) + \exp(\mathbf{X}\hat{\boldsymbol{\alpha}}^T)\}] \quad (5.5)$$

$\mathbf{A}_3$  is a  $q \times q$  symmetrical matrix. It corresponds to the (negative of the) outcome model estimation equations (second row onwards in Equation 5.1) averaged over all individuals being differentiated with respect to the survival coefficients  $\boldsymbol{\beta}$  (that is, the negative Hessian matrix). This is specific to the survival model chosen and will be demonstrated in Section 5.4.5 for a Royston-Parmar model.

### Stabilised weights

For stabilised weights,  $\mathbf{A}$  is a  $(p+q+2) \times (p+q+2)$  matrix. It can be split into the following sections where  $\mathbf{0}_{a \times b}$  is an  $a \times b$  matrix of zeros:

$$\begin{pmatrix} \mathbf{A}_1 & \mathbf{0}_{p+1 \times 1} & \mathbf{0}_{p+1 \times q} \\ \mathbf{0}_{1 \times p+1} & A_4 & \mathbf{0}_{1 \times q} \\ \mathbf{A}_5 & \mathbf{A}_6 & \mathbf{A}_3 \end{pmatrix}$$

$\mathbf{A}_1$  is a  $(p+1) \times (p+1)$  matrix and is the same as defined in Equation 5.4.

$\mathbf{A}_3$  is a  $q \times q$  matrix and is the same as defined for the unstabilised weights but with stabilised weights instead.

$A_4$  is a scalar and corresponds to the (negative of the) estimating equation (second row in Equation 5.2) averaged over all individuals being differentiated with respect to  $\alpha_s$ . This can be estimated as follows:

$$A_4 = \frac{\exp(\hat{\alpha}_s)}{\{1 + \exp(\hat{\alpha}_s)\}^2} \quad (5.6)$$

$\mathbf{A}_5$  is a  $q \times (p+1)$  matrix and only differs from  $\mathbf{A}_2$  because stabilised weights

were used.  $\mathbf{A}_5$  can be estimated as follows:

$$\begin{aligned}\mathbf{A}_5 = & \frac{-1}{n} \left\{ \frac{\partial l^s(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\theta}}} \right\}^T \mathbf{X} \odot \\ & [(1 - \hat{\pi}) \exp(\mathbf{X} \hat{\boldsymbol{\alpha}}^T) - \mathbf{z} \odot \{\hat{\pi} \exp(-\mathbf{X} \hat{\boldsymbol{\alpha}}^T) + (1 - \hat{\pi}) \exp(\mathbf{X} \hat{\boldsymbol{\alpha}}^T)\}]\end{aligned}\quad (5.7)$$

$\mathbf{A}_6$  is a  $q \times 1$  vector. It corresponds to the (negative of the) estimating equations (third row onwards in Equation 5.2) averaged over all individuals being differentiated with respect to  $\alpha_s$ . This can be estimated as follows:

$$\mathbf{A}_6 = \frac{-1}{n} \left\{ \frac{\partial l^s(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} \Big|_{\hat{\boldsymbol{\theta}}} \right\}^T \left[ \frac{\mathbf{z} \exp(-\hat{\alpha}_s)}{\hat{e}(\mathbf{X}) \{1 + \exp(-\hat{\alpha}_s)\}^2} - \frac{(\mathbf{1} - \mathbf{z}) \exp(\hat{\alpha}_s)}{\{\mathbf{1} - \hat{e}(\mathbf{X})\} \{1 + \exp(\hat{\alpha}_s)\}^2} \right] \quad (5.8)$$

Equation 5.3 gives the corrected variance-covariance matrix, which can then be used for various model predictions, for example, using the delta method.

### 5.4.5 M-estimation Variance Estimator for an IP Weighted Royston-Parmar Model

Formulas for a Royston-Parmar survival outcome model are demonstrated in this section. For simplicity, the treatment variable is modelled time-independently, but the methods easily extend to time-dependent effects. Formulas for an exponential, Weibull, Gompertz, log-logistic and log-normal IP weighted survival model are given in Appendix G.1. Where applicable, the formulas in Appendix G.1 can also easily be extended to model treatment as part of the ancillary parameter (not shown, but included in the command `stipw`).

Recall the description and formulas for a Royston-Parmar model from Section 2.3.4. In particular, the log-likelihood in the setting of time-independent effects was given in Equation 2.14. This is repeated below but assuming one covariate

(treatment) with the corresponding coefficient  $\beta_1$ .

$$l_i(\beta_1, \gamma | t_i, \delta_i, z_i, \mathbf{k}_0) = \delta_i (\log [g' \{\log(t_i) | \gamma, \mathbf{k}_0\}] + \eta_i(t_i)) - \exp \{\eta_i(t_i)\}$$

$$\eta_i(t_i) = g \{\log(t_i) | \gamma, \mathbf{k}_0\} + \beta_1 z_i$$

Following the notation in this chapter,  $\boldsymbol{\beta} = (\beta_1, \beta_2 = \gamma_1, \beta_3 = \gamma_2, \dots, \beta_K = \gamma_{K-1}, \beta_{K+1} = \gamma_0)$ . Following Equation 2.25, the weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,  $s_i$ , as defined in Equation 2.24:

$$l_i^w(\boldsymbol{\beta} | t_i, \delta_i, z_i, w_i, \mathbf{k}_0) = w_i \delta_i (\log [g' \{\log(t_i) | \beta_2, \beta_3, \dots, \beta_{K+1}, \mathbf{k}_0\}] + \eta_i(t_i)) -$$

$$w_i \exp \{\eta_i(t_i)\} \quad w_i = s_i, u_i$$

In the following equations, the functional dependencies of  $\eta_i$  ( $T_i$  or  $t_i$ , as appropriate) and the functional dependencies of  $g'$  have been omitted from the notation for ease of exposition. The dependency on the knots  $\mathbf{k}_0$  has also been omitted for ease of exposition. The estimating equations from Equation 5.1 for unstabilised weights become the following. The estimating equations from Equation 5.2 can be written similarly for stabilised weights.

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ u_i Z_i \{\delta_i - \exp(\eta_i)\} \\ u_i [v_{1i} \{\delta_i - \exp(\eta_i)\} + \delta_i v'_{1i} (g')^{-1}] \\ \vdots \\ u_i [v_{K-1i} \{\delta_i - \exp(\eta_i)\} + \delta_i v'_{K-1i} (g')^{-1}] \\ u_i \{\delta_i - \exp(\eta_i)\} \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights is estimated as follows (similar for stabilised weights). Note that some elements of

the matrix are split onto two lines for ease of viewing.

$$\mathbf{A}_3 = \widehat{u}_i \begin{pmatrix} z_i^2 \exp(\widehat{\eta}_i) & & & \\ z_i v_{1i} \exp(\widehat{\eta}_i) & v_{1i}^2 \exp(\widehat{\eta}_i) + & & \\ & \delta_i v'_{1i}^2 (\widehat{g}')^{-2} & & \\ \vdots & \vdots & \ddots & \\ z_i v_{K-1i} \exp(\widehat{\eta}_i) & v_{1i} v_{K-1i} \exp(\widehat{\eta}_i) + \dots & v_{K-1i}^2 \exp(\widehat{\eta}_i) + & \\ & \delta_i v'_{1i} v'_{K-1i} (\widehat{g}')^{-2} & \delta_i v'_{K-1i}^2 (\widehat{g}')^{-2} & \\ z_i \exp(\widehat{\eta}_i) & v_{1i} \exp(\widehat{\eta}_i) & \dots & v_{K-1i} \exp(\widehat{\eta}_i) \exp(\widehat{\eta}_i) \end{pmatrix}$$

As discussed, the variance of useful marginal estimands, such as marginal survival probabilities and differences in RMST, can be obtained using the delta method.

## 5.5 Simulation Study Methods

A simulation study was performed to evaluate the performance of the proposed M-estimation variance estimator in comparison to the bootstrap and robust variance estimators. The methods are presented following the ADEMP structure (aims, data generating mechanism, estimands, methods and performance measures) [4].

### 5.5.1 Aims

1. To evaluate the performance of the M-estimation variance estimator (for both stabilised and unstabilised weights).
2. To compare whether stabilised or unstabilised weights result in an M-estimation variance estimator that is closest to the corresponding empirical variance. See Section 5.5.5 for more details on the performance measure used to evaluate Aims 1 and 2.
3. To compare the computational time of the M-estimation and bootstrap variance estimators.

## 5.5.2 Data Generating Mechanism

### General Algorithm

The data generating mechanism was the same as that described in Section 4.5.2.

### Evaluated Scenarios

The following parameters were varied in the simulation study:

- Parameters of the marginal Weibull outcome model:  $\{\lambda, \gamma\} = \{0.15, 0.7\}, \{0.003, 1.4\}$
- Treatment prevalence:  $\pi_Z = \{0.1, 0.25, 0.5\}$
- Treatment effect:  $\exp(\beta) = \{0.5, 2\}$
- No intermittent censoring and intermittent censoring rate parameter  $\lambda_C = 0.05$
- Sample size:  $n_{obs} = \{200, 10000\}$

This led to a total of  $2 \times 3 \times 2 \times 2 \times 2 = 48$  scenarios. An additional set of Weibull parameters was added (compared to the simulation study in Chapter 4) to investigate the case of a low event rate. The number of treatment effects investigated was reduced to strong effects (in each direction) as these were deemed sufficient to investigate the impact of treatment effect and would help reduce the computational time.

Two sample sizes were considered:  $n_{obs} = \{200, 10000\}$ . The focus of the simulation study is on the large sample properties of the proposed variance estimator. The iteration sample size calculation, exploratory results and main results are all reported for  $n_{obs} = 10000$  only. Section 5.6.3 will discuss the exploratory results of the small sample size.

### Simulation Parameters

Unless being varied across the scenarios, all other parameter values were the same as detailed in Section 4.5.2.

### 5.5.3 Estimands

As in Section 4.5.3, two estimands were of interest:

1. The marginal log hazard ratio  $\beta$
2. The difference in the marginal RMST at time  $t = 20$ ,  $\Delta_\mu(20)$

### 5.5.4 Methods

For all methods, the same IP weighted Weibull survival model was fitted to the data (following the methods described in Section 2.6). Therefore, all three methods gave the same point estimate. The following variance estimators were then used:

1. **Robust variance estimator:** see Section 5.3.2
2. **Bootstrap variance estimator:** with  $m = 500$  bootstrap samples, see Section 5.3.3
3. **M-estimation variance estimator:** see Section 5.4

Each approach was used for both stabilised and unstabilised weights resulting in 6 methods in total.

### 5.5.5 Performance Measures

The notation in Table 3.4 is used here. The empirical standard error (EmpSE) is defined as  $\sqrt{\text{Var}(\hat{\theta})}$  while the average model-based standard error (ModSE) is defined as  $\sqrt{E\{\widehat{\text{Var}}(\hat{\theta})\}}$  [4]. The EmpSE estimates the standard deviation of  $\hat{\theta}_i$  over the  $n_{sim}$  iterations [4]. In this simulation study, for a given weight (stabilised/unstabilised), the EmpSE will be the same for all three methods (as the point estimates  $\hat{\theta}_i$  are the same for each method). Alternatively, the ModSE takes the average of the variance estimated in each iteration  $i$  and then takes the square root. If a variance estimator performs well, the ModSE should be roughly equal to the EmpSE. The following performance outcomes were analysed:

1. **Aims 1-2:** The relative percentage error in ModSE:  $100(\frac{\text{ModSE}}{\text{EmpSE}} - 1)$  and corresponding MCSE, as given in Equation 3.3.
2. **Aim 3:** The average computational time. The time taken to estimate the variance for both weights, estimands and sample sizes is recorded for each repetition in each scenario.

### Iteration Sample Size

An iteration sample size calculation was performed to ensure the relative percentage error in ModSE was estimated to an acceptable degree of precision [4]. Equation 3.3 can be rearranged to give the iteration sample size for a given value of MCSE:

$$n_{sim} = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (5.9)$$

$$a = \text{MCSE}^2$$

$$b = - \left( \text{MCSE}^2 + \frac{100^2 \widehat{\text{ModSE}}^2}{\widehat{\text{EmpSE}}^2} \left[ \frac{\widehat{\text{Var}} \{ \widehat{\text{Var}} (\widehat{\theta}) \}}{4 \widehat{\text{ModSE}}^4} + \frac{1}{2} \right] \right)$$

$$c = \frac{100^2 \widehat{\text{Var}} \{ \widehat{\text{Var}} (\widehat{\theta}) \}}{4 \widehat{\text{EmpSE}}^2 \widehat{\text{ModSE}}^2}$$

A maximum MCSE value of 1 percentage point for both estimands was deemed acceptable. Estimates of ModSE, EmpSE and  $\widehat{\text{Var}} \{ \widehat{\text{Var}} (\widehat{\theta}) \}$  were calculated by performing the simulation with 1000 iterations. Estimates were for each of the estimands, for each method/weight and for each of the 24 scenarios for the sample size 10000. These estimates were plugged into the equation above to obtain the iteration sample size needed. The maximum over the scenarios (sample size 10000 only) and methods/weights was taken for each estimand and the iteration sample size for the main study was chosen as the maximum across the estimands. The results are shown in Table 5.1. The iteration sample size was determined to be 7700.

**Table 5.1:** Estimated  $n_{sim}$  required so that the MCSE for the relative percentage error in the model-based standard error is less than or equal 1 percentage point and the estimated maximum MCSE when  $n_{sim} = 7700$  (for the 10000 sample size)

Estimand	$n_{sim}$ needed	Maximum MCSE if $n_{sim} = 7700$
Marginal log hazard ratio	7151	0.96
<b>Difference in marginal RMST</b>	<b>7680</b>	<b>1.00</b>

### 5.5.6 Software

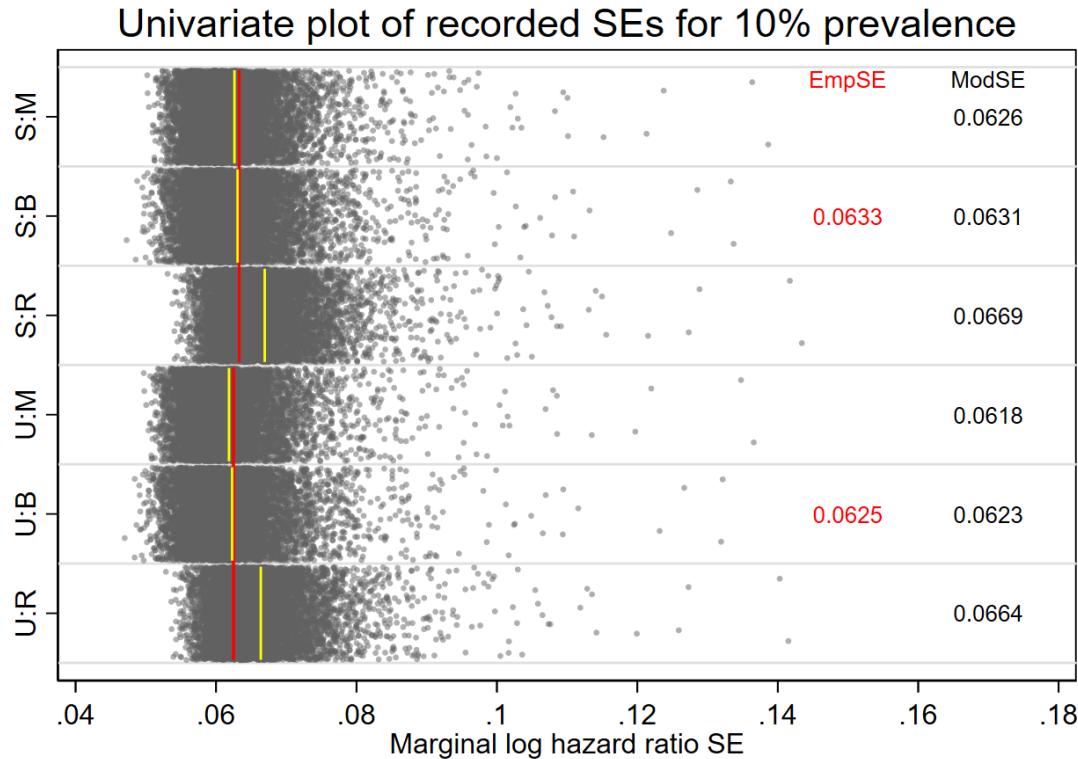
All simulations were performed in **Stata/MP** 4-core, version 17. The newly written command **stipw** was used for the robust and M-estimation variance estimators for an IP weighted Weibull model, see Chapter 6. **stipw** was programmed to use a Royston-Parmar model with 1 degree of freedom (with the **noorthog** option specified), which is equivalent to a Weibull model programmed with **streg**. For bootstrapping, **streg** was used and RMST was calculated directly using numerical integration with **integ** rather than with a postestimation command (as bootstrapping only requires point estimates and this approach is equivalent and quicker).

The simulation was ran in 24 batches on the University of Leicester’s High Performance Computing cluster. Each batch ran the simulation for the corresponding 200 and 10000 sample sizes for each scenario. In total, two simulations were performed: a preliminary simulation used in the iteration sample size calculation ( $n_{sim} = 1000$ ) and the main simulation study ( $n_{sim} = 7700$ ). Starting seeds were varied for each simulation and each batch and are given in Table G.1.

## 5.6 Simulation Study Results

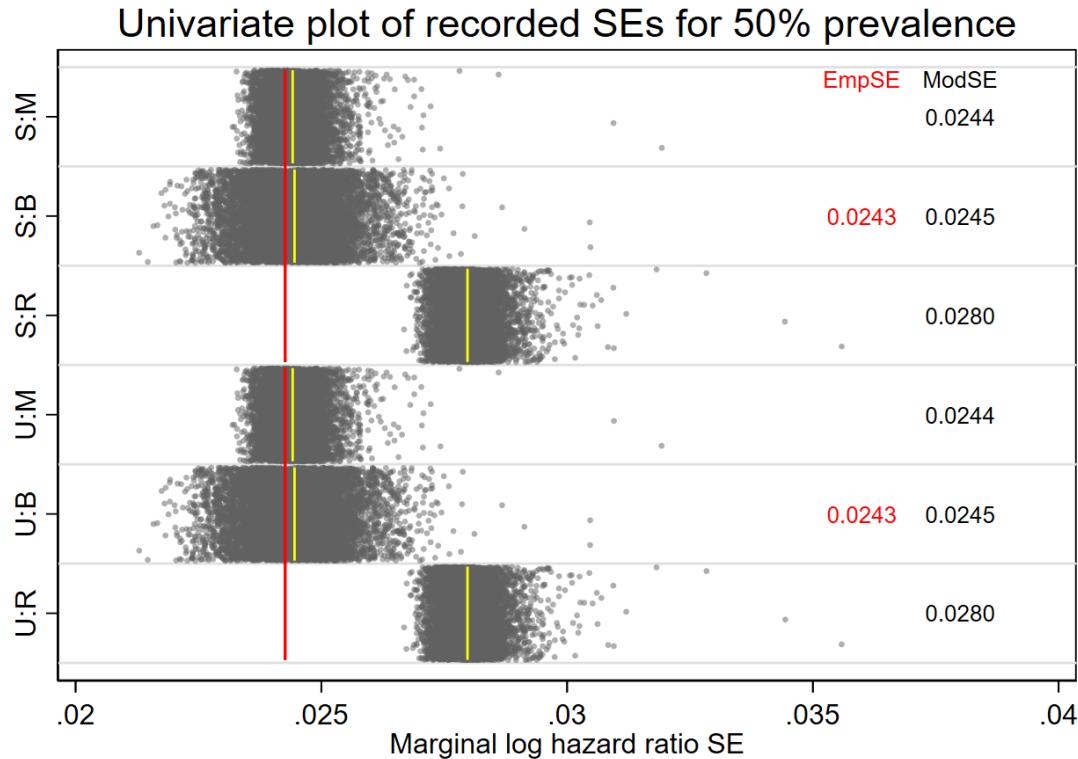
### 5.6.1 Exploratory Analysis

This and the following subsection consider only the large sample size ( $n_{obs} = 10000$ ). All models converged in all scenarios. Exploratory graphs showed a similar spread of the model-based standard errors across the methods when the treatment preva-



**Figure 5.1:** Univariate plot showing the spread of the model-based standard errors for the marginal log hazard ratio when  $\gamma = 0.7$ , the treatment prevalence  $\pi_Z = 0.1$ , the marginal hazard ratio  $\exp(\beta) = 0.5$  and there is no intermittent censoring for the large sample size  $n_{obs} = 10000$ . U:R, U:B and U:M are the robust, bootstrap and M-estimation variance estimators with unstabilised weights, respectively. S:R, S:B and S:M are the corresponding variance estimators with stabilised weights. The vertical red lines represent the empirical standard error for each weight, see Section 5.5.5. The value is given on the right-hand side in red text. The vertical yellow lines represent the average model-based standard error for each method. The value is given on the right-hand side in black text

lence was 10%, with the robust variance estimator often slightly overestimating the empirical standard error, for example, see Figure 5.1. As the prevalence increased, generally, the spread was smaller for M-estimation relative to bootstrapping, although both gave model-based standard errors close to the empirical standard error. The robust variance estimator gave increasingly more conservative estimates as the prevalence increased to 50%, for example, see Figure 5.2. There was often good concordance between the stabilised and unstabilised model-based standard errors for M-estimation.



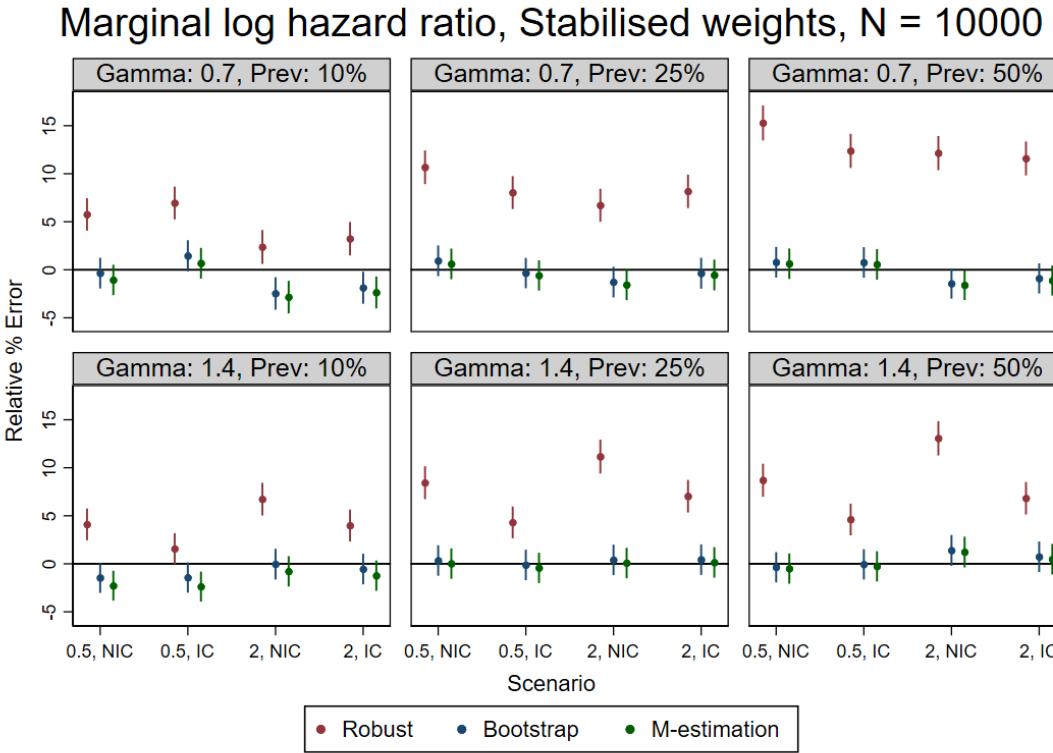
**Figure 5.2:** Univariate plot showing the spread of the model-based standard errors for the marginal log hazard ratio when  $\gamma = 0.7$ , the treatment prevalence  $\pi_Z = 0.5$ , the marginal hazard ratio  $\exp(\beta) = 0.5$  and there is no intermittent censoring for the large sample size  $n_{obs} = 10000$ . U:R, U:B and U:M are the robust, bootstrap and M-estimation variance estimators with unstabilised weights, respectively. S:R, S:B and S:M are the corresponding variance estimators with stabilised weights. The vertical red lines represent the empirical standard error for each weight, see Section 5.5.5. The value is given on the right-hand side in red text. The vertical yellow lines represent the average model-based standard error for each method. The value is given on the right-hand side in black text

### 5.6.2 Main Analysis

The maximum estimated MCSE for the relative percentage error was 0.93 and 0.94 percentage points for the marginal log hazard ratio and difference in marginal RMST for  $n_{obs} = 10000$ , respectively. These were below the maximum acceptable MCSE threshold of 1 percentage point, as specified in Section 5.5.5, and suggests the iteration sample size was sufficient. The average MCSE across the scenarios and methods/weights was 0.83 and 0.84 for the estimands, respectively.

## Aim 1: Performance of the M-estimation Variance Estimator

Figures 5.3 and 5.4 show the relative percentage error with confidence intervals for the marginal log hazard ratio and difference in marginal RMST, respectively. The corresponding values are given in Tables G.2-G.5 in the Appendix.

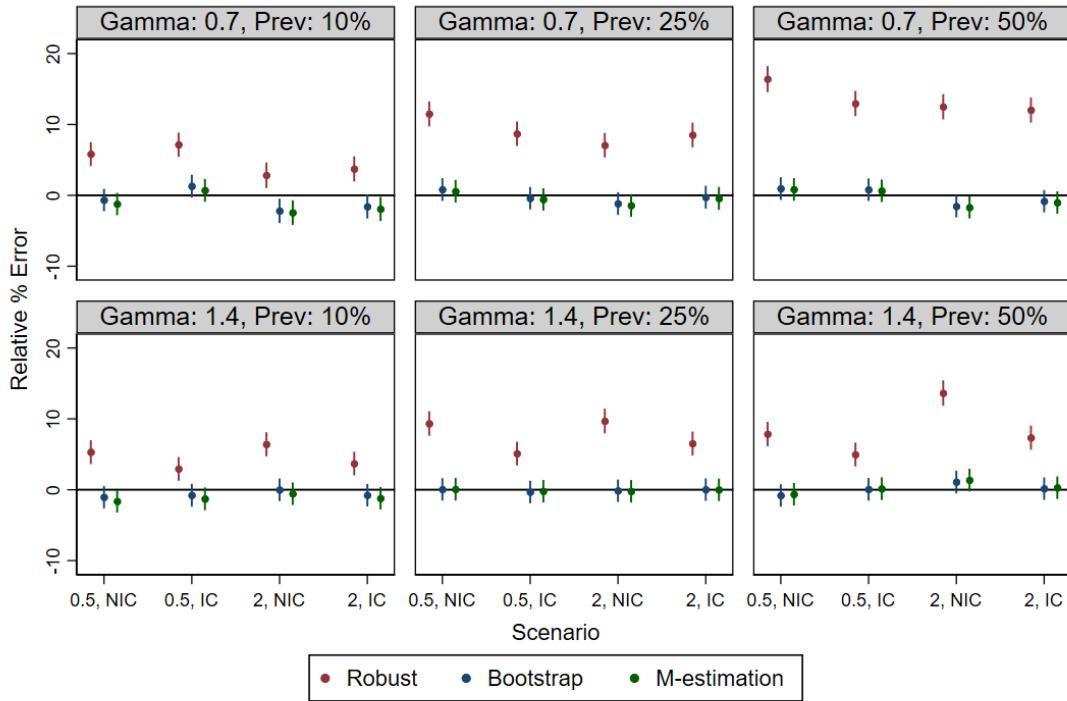


**Figure 5.3:** The relative percentage error of the stabilised variance estimators for the marginal log hazard ratio for the large sample size  $n_{obs} = 10000$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

Overall, both the M-estimation and bootstrap variance estimators performed well, giving relative errors close to 0. In most cases, M-estimation gave slightly smaller model-based standard errors compared to bootstrapping, although the difference between M-estimation and bootstrapping reduced as the treatment prevalence increased. In general, the M-estimation estimator had the greatest absolute

percentage error for the smallest treatment prevalence. The figures show that the robust variance estimator was too conservative, with the relative error increasing as the treatment prevalence increased. The figures in the main text are for stabilised weights; unstabilised weights gave similar results and the corresponding graphs are shown in Figures G.1 and G.2 in the Appendix.

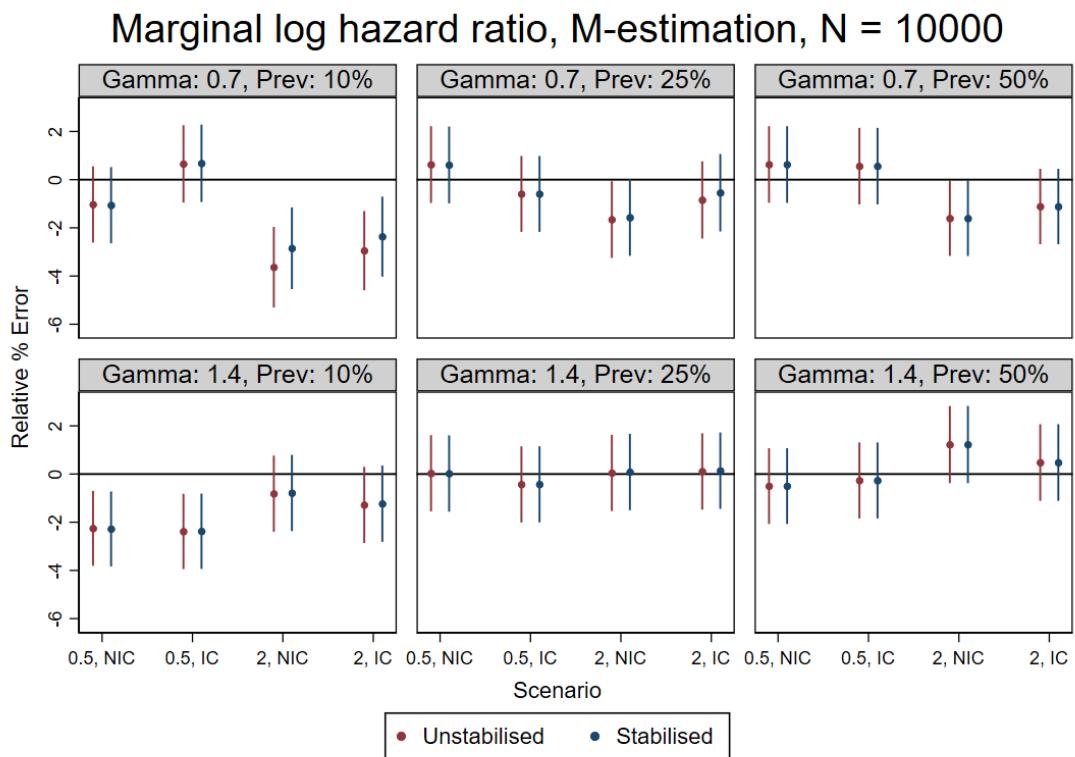
### Difference in marginal RMST, Stabilised weights, $N = 10000$



**Figure 5.4:** The relative percentage error of the stabilised variance estimators for the difference in marginal RMST for the large sample size  $n_{obs} = 10000$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

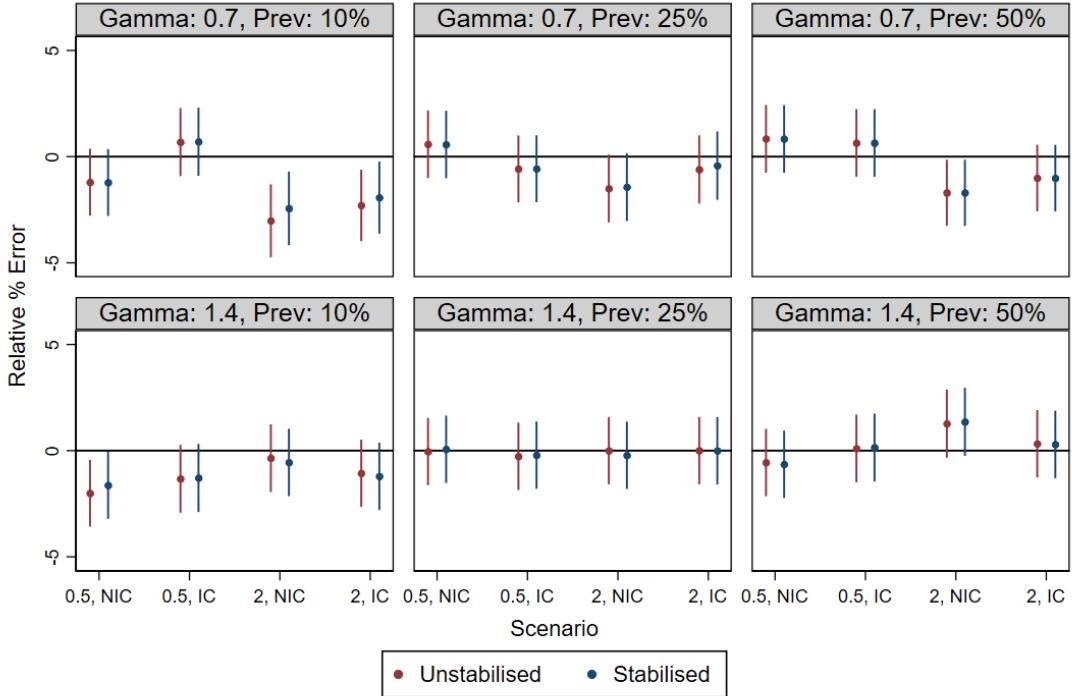
## Aim 2: Comparison of Stabilised and Unstabilised Weights with the M-estimation Variance Estimator

Figures 5.5 and 5.6 show the relative percentage error with confidence intervals for the marginal log hazard ratio and difference in marginal RMST, respectively, with stabilised and unstabilised M-estimation variance estimates. The M-estimation variance estimates were very similar for the two weights. In two scenarios was there a slight difference ( $\gamma = 0.7$ ,  $\pi_Z = 0.1$  and  $\exp(\beta) = 2$ ); however, the difference was unlikely to be of practical importance.



**Figure 5.5:** The relative percentage error of the M-estimation variance estimators for the marginal log hazard ratio for the large sample size  $n_{obs} = 10000$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

## Difference in marginal RMST, M-estimation, N = 10000



**Figure 5.6:** The relative percentage error of the M-estimation variance estimators for the difference in marginal RMST for the large sample size  $n_{obs} = 10000$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

### Aim 3: Comparison of Computational Time

The computational time (hours) to calculate the marginal log hazard ratio and difference in marginal RMST for both sample sizes for all methods/weights was recorded for each of the 24 scenarios (batches). The bootstrapping time does not include calculating the point estimate and instead just includes the bootstrap simulation. The summary data across the 24 scenarios are given in Table 5.2. As expected, M-estimation was considerably quicker than bootstrapping. At best, in one scenario M-estimation took 2.3% of the time of bootstrapping and, at worst,

M-estimation took 7.9% of the bootstrapping time. On average, M-estimation took 5% of the bootstrapping time.

**Table 5.2:** Computational time (hours) to calculate the marginal log hazard ratio and difference in marginal RMST for both sample sizes for both weights in the simulation study. The table gives summary data across the 24 scenarios (batches)

24 scenarios	M-estimation (hours)	Bootstrap (hours)	M-estimation/ Bootstrap (%)
Mean	18.0	350.7	5.2
(SD)	(6.2)	(26.3)	(1.9)
Median	18.6	336.3	5.4
(LQ, UQ)	(12.3, 23.3)	(334.8, 370.0)	(3.5, 6.7)
Min, Max	8.9, 28.0	320.5, 404.4	2.3, 7.9

### 5.6.3 Exploratory Analysis for the Small Sample Size

Although the simulation study was designed to investigate the large sample properties of the proposed variance estimator, a sample size of 200 was also considered as an exploratory analysis. The simulation study was not designed for small samples and some difficulties were encountered. Some were due to few individuals being assigned to one of the treatment groups and/or no events in one of the treatment groups. In rare cases, the former difficulty led to issues with the bootstrap variance estimator; where extreme bootstrap samples resulted in non-convergence of the logistic regression model or the propensity score being estimated as 0 or 1 (and the weights being undefined). The latter difficulty was uncommon and, in such situations, alternate analyses should be employed. There was a further difficulty with the bootstrap variance estimator when there were few events in one of the treatment groups, as extreme bootstrap samples resulted in considerably large variance estimates.

The exploratory results from the small sample simulation have been included in Appendix G: Figures G.3-G.6 and Tables G.6-G.9, excluding the estimates with the aforementioned difficulties. In most scenarios, the relative percentage error was not close to 0 for most variance estimators. In general, the empirical standard

error was underestimated by M-estimation, overestimated by bootstrapping and was underestimated/overestimated by the robust variance estimator for 10%/50% treatment prevalence, respectively. Further work is needed to investigate the small sample properties of the proposed variance estimator.

## 5.7 Illustrative Examples

### 5.7.1 General Methods

Three datasets were analysed to demonstrate the performance of the M-estimation variance estimator, compared to the robust and bootstrap variance estimators. A brief data description is given in each subsection, including how the treatment/ exposure variable was modelled and modelling details specific to that dataset. Standardised differences in the raw and weighted data were reviewed. For the STD dataset, a weighted Weibull model was used for continuity with Section 4.3. For the other two datasets, the weighted survival model was determined as the overall best performing distribution according to AIC and BIC across both stabilised and unstabilised weights. The options included: exponential, Weibull, Gompertz, log-normal, log-logistic and Royston-Parmar models with 1-5 degrees of freedom each with no and 1-5 degrees of freedom for the treatment/exposure variable.

Robust, bootstrap and M-estimation standard errors were obtained from the chosen IP weighted survival model using both stabilised and unstabilised weights. The estimand of interest was the difference in marginal RMST (at a time point specific to each dataset). For continuity, the marginal hazard ratio was also investigated for the STD dataset. Robust and bootstrap standard errors were calculated, with the latter using  $m = 500$  (same as the simulation study) and  $m = 10000$  bootstrap samples.  $m = 10000$  was chosen because a considerably larger number of bootstrap samples is often used outside of simulation studies, when a single analysis is performed. M-estimation standard errors were calculated using the proposed variance estimator, as discussed in Section 5.4. The computational time was also reported. The analysis details regarding software are the same as those for the simulation

study, see Section 5.5.6.

### 5.7.2 STD Dataset

The analysis of the STD dataset in Section 4.3 was extended. 264/585 (45.1%) and 83/292 (28.4%) of Black and White individuals were reinfected, respectively. The  $\gamma$  and  $\lambda$  parameters were estimated to be 0.74 and 0.40 in the fitted, IP weighted Weibull model with stabilised weights, respectively. The parameter estimates were similar for the unstabilised weights. As given in Table 4.2, the marginal hazard ratio predicted by the IP weighted Weibull model was 1.14 and the difference in marginal RMST at 4 years was -0.16 years for both weight types.

The estimated standard errors using the different approaches are given in Table 5.3. M-estimation gave slightly smaller standard errors than bootstrapping with 10000 samples, while robust standard errors appeared to be more consistent with the bootstrap standard errors. The robust and M-estimation standard errors took considerably less time to calculate than the bootstrap standard errors, see Table 5.4.

**Table 5.3:** Standard errors for the marginal hazard ratio and difference in marginal RMST at 4 years for the different variance estimators and types of weights for the STD dataset

STD Dataset	Marginal HR		RMST (4 years)	
	UW	SW	UW	SW
Robust	0.1718	0.1722	0.1844	0.1852
Bootstrap (500)	0.1745	0.1826	0.1811	0.1904
Bootstrap (10000)	0.1797	0.1808	0.1839	0.1845
M-estimation	0.1626	0.1630	0.1746	0.1753

### 5.7.3 AIDS Clinical Trials Group Study 175 (ACTG175) Dataset

The ACTG175 dataset was introduced in Section 1.3.1 and was a randomised clinical trial comparing monotherapy with zidovudine or didanosine with combination therapy with zidovudine and didanosine or zidovudine and zalcitabine [6]. The out-

**Table 5.4:** Computational time (seconds) to calculate the difference in marginal RMST for the different variance estimators and for the different datasets. Results are for stabilised weights

Method	STD Dataset	ACTG175 Dataset	RHC Dataset
Robust	2	5	14
Bootstrap (500)	134	388	708
Bootstrap (10000)	2510	8618	15228
M-estimation	2	5	14

come was years to the first occurrence of (i) a decline CD4 T cell count of at least 50, (ii) an event indicating progression to AIDS or (iii) death [6]. The ACTG175 dataset consists of 2139 individuals and 17 covariates. Although IP weighting is typically applied to observational data, it has been suggested than IP weighting can be used in randomised trials to reduce the variance of the estimated treatment effect [82]. Therefore, despite being a randomised clinical trial, IP weighting has been applied to the ACTG175 dataset in this illustrative example.

Similar to a previous application [163], treatment was recoded to be binary and compared monotherapy (zidovudine alone or didanosine alone) with combined therapy (zidovudine and didanosine or zidovudine and zalcitabine). Of the 2139 individuals, 1093 (51.1%) received monotherapy. A logistic regression model was used to model treatment incorporating 15 of the possible confounders. **zprior** (zidovudine use prior to treatment initiation) was excluded as it was singular and **str2** (antiretroviral history) was excluded as it was collinear with **strat** (antiretroviral history stratification).

The standardised differences for the raw data showed good balance between the groups (maximum standardised difference of 0.064), which is to be expected from a randomised clinical trial. Applying weights (both types) decreased the maximum to 0.034. The unstabilised weights ranged from 1.49 to 2.74 with mean 2.00 and median 1.99. In comparison, the stabilised weights ranged from 0.76 to 1.34 with mean and median 1.00.

In total, the analysis included 5148 person-years and the maximum follow-up

was 3.4 years. 521 events (24.4%) were experienced with 309/1093 (28.3%) and 212/1046 (20.3%) experienced in the monotherapy and combined therapy groups, respectively. An IP weighted Royston-Parmar model with 2 degrees of freedom and 1 degree of freedom for treatment was used, as this model gave consistently low AIC and BIC values across the stabilised and unstabilised weights ( $5^{th}$  lowest AIC and  $1^{st}$  lowest BIC for unstabilised and  $2^{nd}$  lowest AIC and  $4^{th}$  lowest BIC for stabilised).

The estimated difference in marginal RMST at three years was 0.1649 for both unstabilised and stabilised weights, suggesting that at three years individuals on the combination therapy were event-free for 60 more days on average than individuals on monotherapy. An unweighted analysis gave a difference in marginal RMST of 0.1659 (naive standard error: 0.03105), which was similar to its IP weighted counterpart.

The estimated standard errors using the different approaches are given in Table 5.5. M-estimation gave standard errors consistent with bootstrap standard errors (10000 samples) and were both smaller than the robust standard errors. Compared to the naive standard error of the unweighted analysis, both bootstrapping and M-estimation appeared to offer increased precision. The timings of the analysis for stabilised weights are shown in Table 5.4. M-estimation was considerably quicker and took 1.3% and 0.06% of the bootstrap computational time with 500 and 10000 samples, respectively.

**Table 5.5:** Standard errors for the difference in marginal RMST at 3 years for the different variance estimators and types of weights for the ACTG175 dataset

ACTG175 Dataset	Unstabilised	Stabilised
Point estimate	0.1649	0.1649
Robust	0.03138	0.03083
Bootstrap (500)	0.02917	0.02997
Bootstrap (10000)	0.02980	0.02966
M-estimation	0.02988	0.02936

### 5.7.4 Right Heart Catheterisation (RHC) Dataset

The RHC dataset was introduced in Section 1.3.4 and was an observational study performed to investigate the effectiveness of RHC in the initial care of critically ill patients. The study compared patients receiving a RHC within 24 hours of admission to those who did not in hospitalised adult patients. The outcome was time to death (in years). The RHC dataset consists of 5735 individuals and 53 covariates (73 dummy variables).

Of the 5509 included individuals (exclusions explained in this paragraph), 2085 (37.9%) received a RHC within 24 hours of admission. A logistic regression model was used to model RHC incorporating 49 of the possible confounders (67 dummy variables) in 5509 individuals. Covariates with more than 5% missing values (or continuous variables with illogical 0 entries, assumed to be missing) were excluded. This resulted in the variables `urin1` (urine output), `adld3pc` (Activities of Daily Life) and `wtkilo1` (weight) being excluded. Observations with missing values (or illogical 0 entries) for covariates under this threshold were dropped leaving 5509 individuals. Lung and colon cancer were grouped in `cat1` (primary disease category) and `cat2` (secondary disease category), due to small numbers and both disease types being cancer. `surv2md1` (SUPPORT model estimate of the probability of surviving two months) was not included, as this was a prediction of survival rather than a baseline characteristic. This resulted in 49 covariates (67 dummy variables).

The standardised differences for the raw data showed very poor balance between the groups (maximum standardised difference of 0.599). Of the 49 covariates, 30 had a standardised difference greater than 0.1 with 4 being greater than 0.4. Applying weights (both types) improved the balance, although some covariates still had poor balance. 20 covariates had a standardised difference greater than 0.1 and the maximum was 0.379. The unstabilised weights ranged from 1.00 to 41.00 with mean 1.98 and median 1.45. In comparison, the stabilised weights ranged from 0.39 to 15.52 with mean 0.99 and median 0.78.

In total, the analysis included 2844 person-years and the maximum follow-up was 5.3 years. There were 3549 deaths (64.4%), 1410/2085 (67.6%) in the RHC group

and 2139/3424 (62.5%) in the no RHC group. An IP weighted Royston-Parmar model with 5 degrees of freedom and 2 degrees of freedom for RHC within 24 hours was used, as this model gave consistently low AIC and BIC values across the stabilised and unstabilised weights ( $6^{th}$  lowest AIC and  $5^{th}$  lowest BIC for unstabilised and  $1^{st}$  lowest AIC and  $3^{rd}$  lowest BIC for stabilised).

The estimated difference in marginal RMST at 5 years was -0.1030 and -0.1026 for the unstabilised and stabilised weights, respectively, suggesting that at 5 years individuals who received an RHC within 24 hours lived on average 38 days less than those who did not. This result was surprising and may indicate that the propensity score (treatment model) was not correctly specified. Further evidence of model misspecification was that there was still imbalance between the groups for some covariates in the weighted dataset.

The estimated standard errors using the different approaches are given in Table 5.6. For both weights, the M-estimation standard errors were smallest, followed by the bootstrap standard errors for 10000 samples and then the robust standard errors. The timings of the analysis for stabilised weights are shown in Table 5.4. M-estimation took 2.0% and 0.09% of the bootstrap computational time with 500 and 10000 samples, respectively.

**Table 5.6:** Standard errors for the difference in marginal RMST at 5 years for the different variance estimators and types of weights for the RHC dataset

RHC Dataset	Unstabilised	Stabilised
Point estimate	-0.1030	-0.1026
Robust	0.05286	0.0529
Bootstrap (500)	0.05131	0.04774
Bootstrap (10000)	0.05148	0.05203
M-estimation	0.04963	0.04965

## 5.8 Discussion

This chapter proposed a closed-form variance estimator for a range of IP weighted parametric survival models, which utilises M-estimation to correctly account for

the associated uncertainty in the weight estimation. The variance of useful causal estimands, such as marginal survival probabilities or RMST, can be subsequently obtained using the delta method. The performance of the proposed variance estimator was evaluated in a simulation study, which compared the relative percentage error of the proposed variance estimator to the bootstrap and robust variance estimators. The proposed variance estimator was illustrated on three datasets using a newly written **Stata** command, **stipw**, which is described in the next chapter.

The simulation study demonstrated that for large sample sizes, the proposed variance estimator and bootstrap variance estimator performed well, giving relative percentage errors close to 0. M-estimation had the greatest absolute errors for the smallest treatment prevalence; a trend also observed by Hajage *et al* [2] for their closed-form variance estimator. The robust variance estimator gave overly conservative estimates, especially as the treatment prevalence increased towards 50%, as demonstrated by Austin [19]. The proposed variance estimator could be an alternative to bootstrapping in scenarios similar to those explored in the simulation study.

The M-estimation variance estimator with stabilised and unstabilised weights gave similar estimates. In two of the more extreme scenarios, stabilised weights offered a slight advantage over unstabilised weights; however, the difference was unlikely to be of practical importance. While in marginal structural models and models with continuous treatments, stabilised weights can provide smaller variance estimates; this result was not evident in this simulation study. Extreme weights may be more common in marginal structural models and this may explain why not much benefit was gained with the stabilised M-estimation variance estimator compared to its unstabilised counterpart. Had scenarios more prone to extreme weights been investigated, for example, an even smaller treatment prevalence, then the findings of the simulation study may have been more in line with the recommendation for marginal structural models.

As discussed by Hajage *et al* [2], M-estimation (as a closed-form variance estimator) has the advantage of being more easily reproducible (it does not require a num-

ber of samples, starting seed or knowledge of the random number generator) and is considerably quicker than bootstrapping. On average, the M-estimation predictions took 5% of the time of the corresponding bootstrap estimates with 500 samples in the simulation study. This could have been reduced further if `standsurv` was used instead of `predictnl` to obtain the difference in marginal RMST with standard errors for M-estimation. M-estimation could, therefore, be an especially useful tool for large datasets when multiple analyses are required (for example, exploratory analyses, choosing the propensity score model and/or survival model and for sensitivity analyses) and when repetitive use of bootstrapping may be burdensome.

Closed-form variance estimators for IP weighted survival models have been previously proposed by directly modelling [21], linearising [2] and Poissonising [20] the Cox model. However, alternatives to the hazard ratio, such as marginal survival probabilities and RMST, are growing in popularity. An advantage of providing a closed-form variance estimator for parametric IP weighted survival models is that the variance of these useful marginal predictions can easily be obtained using the delta method. Complex hazard functions can still be accommodated by utilising flexible Royston-Parmar models, which can also allow for non-proportional hazards using time-dependent effects [8, 40, 45].

The proposed variance estimator is similar to that of Mao *et al* [20], as they both use M-estimation with parametric IP weighted survival models; however, there are two key differences. Firstly, this work defines the M-estimation variance estimator for a number of parametric survival models, while Mao *et al* focus on approximating the Cox model. Poissonising the Cox model requires splitting the dataset into intervals defined by the failure times. Analysing the expanded dataset can be computationally intensive, especially as the size of the dataset increases, for example, as is the case in electronic health records. The approach in this thesis is therefore expected to be more efficient.

Secondly, the estimating equations are structured differently. In the proposed approach, all the parameters in the outcome survival model are estimated simultaneously. The variance of the coefficient of treatment (for example, the marginal log

hazard ratio in a proportional hazards model) is directly available and the variance of estimands, such as the marginal survival probabilities or RMST, can be obtained using the delta method. Conversely, Mao *et al* has two sets of estimating equations - one for the marginal hazard ratio and the other for survival probabilities and related estimands. In the latter set, Mao *et al* assumes a separate survival function for each treatment group, while the proposed approach can essentially fit separate survival models (with the use of ancillary parameters) or a single model, if, for example, the proportional hazards assumption is valid.

The approach in this thesis considered two types of weights: unstabilised and stabilised. The latter requires an additional estimating equation to take into account the sampling variability of the prevalence of treatment. Previous closed-form variance estimators for IP weighted survival models have considered alternative weighting strategies, which target different estimands than the average treatment effect in the sampled population. For example, Mao *et al* [20] considered the matching weight [96] and overlap weight [97], while Hajage *et al* [2] considered weights that target the average effect among the treated. The proposed variance estimator could be extended to handle alternate weighting strategies.

The proposed variance estimator assumed a fixed, binary treatment. A second extension of this work could be to allow for more than two treatment groups by including additional estimating equations in the stacked equations, one for each treatment level, and employing multinomial regression to model treatment. Extending the methodology to a continuous or time-varying treatment would require more consideration. Another extension would be to use alternate methods to logistic regression to model the propensity score, such as a probit model, as long as they had a well-defined estimating equation to be included in the stacked estimating equations [21]. Finally, the methodology could be extended to more complex data structures, for example, to allow for clustered data similar to Shu *et al* [21].

As with any simulation study, further scenarios could have been explored. Useful further work would be to extend the simulation study to investigate the influence of the number and strength of the confounder variables and model misspecification in

the treatment model. The variance of the parameters from the treatment model is incorporated into the variance calculation of the parameters in the weighted survival model via M-estimation. Therefore, different magnitudes and/or bias of the variance and variance-covariance matrix structures may affect M-estimation more acutely than they may affect bootstrapping or the robust variance estimator. In addition, it would be useful to explore the robustness of the proposed estimator to departures of the conditional exchangeability and positivity assumptions. Further investigations would be useful and may help explain the slight differences seen in the illustrative datasets.

The simulation study focused on large sample properties and included brief exploratory analyses for the small sample properties. In most small sample scenarios, the relative percentage error was not close to 0 for most methods. In general, the empirical standard error was underestimated by M-estimation, overestimated by bootstrapping and was underestimated/overestimated by the robust variance estimator for 10%/50% treatment prevalence, respectively. In some scenarios of high survival and low treatment prevalence, bootstrapping severely overestimated the variance, as was also demonstrated by Shu *et al* [21].

However, the simulation study design did not appropriately handle issues arising from small sample sizes. For example, there were datasets with zero events in either the treatment or control group, which gave an extremely small/large estimated marginal log hazard ratio. The corresponding marginal RMST (in the group with no events) was estimated with a standard error of 0. These datasets were removed, as if presented with a dataset with no events, alternative methods would be used to analyse it. Furthermore, removing these datasets in the simulation study may have induced bias and affected the MCSE. Alternate methods could have included changing the simulation design to ensure there was at least one event in each group, tweaking the parameters of the simulation study or replacing the iterations when this occurred. In addition, removing the most extreme outliers may also have induced bias. More appropriate simulation study designs specifically for small samples should be used to thoroughly investigate the small sample properties of the proposed

variance estimator, and also the bootstrap variance estimator.

Although the simulation study suggested that on average the bootstrap and M-estimation variance estimator perform similarly, with the robust variance estimator often giving considerably different (larger) variance estimates, this was not evident from all the applications explored in Section 5.7. This may be for a number of reasons. Firstly, this could be due to sampling variability. Secondly, the STD and RHC datasets had treatment/exposure prevalences less/more than 50% and the STD dataset was relatively small. The simulation study suggested that these factors increased the differences between the bootstrap and M-estimation variance estimates. Finally, and probably most importantly, the simulation study guaranteed no model misspecification. In the RHC, for example, there was still poor balance across quite a few of the covariates after weighting, suggesting that the propensity score model may have been misspecified. The impact of model misspecification on the variance estimators was not investigated and, as mentioned earlier, would constitute a useful area of further work.

This work has provided an alternative variance estimator to bootstrapping for IP weighted parametric survival models for large samples and may be advantageous if computational time or reproducibility are key concerns. By providing a variance estimator for parametric IP weighted survival models, the variance of useful causal estimands can easily be obtained via the delta method. The proposed framework corresponds to a range of parametric survival models, including flexible Royston-Parmar models, and can be implemented with the newly developed `stipw` command in **Stata**, which is described in the next chapter.

## 5.9 Conclusion

This chapter extended Chapter 4 by proposing a closed-form variance estimator for a range of IP weighted parametric survival models. The estimator was shown to perform well in large samples in a simulation study and could be used as an alternative to bootstrapping. In particular, the M-estimation variance estimator gives more

easily reproducible results and is considerably quicker than bootstrapping. This could be especially useful for treatment model selection and/or sensitivity analyses in very large datasets, when bootstrapping may be too computationally intensive to perform multiple times. This chapter demonstrated the proposed variance estimator on three datasets using a new **Stata** command, which is described in the next chapter.

# Chapter 6

---

## Software Development for the Closed-form Variance Estimator for Inverse Probability Weighted Parametric Survival Models

---

### 6.1 Outline

Chapter 5 proposed a closed-form variance estimator for a range of IP weighted parametric survival models. This chapter focuses on the accompanying software. A new **Stata** command, **stipw**, was written as part of the work in this thesis to obtain the proposed variance estimator. This chapter begins by reviewing related software in **R** and **Stata**. The algorithm of **stipw** is discussed, followed by an explanation of the options available. Example code is then given, which relates to the analysis of the illustrative examples in Section 5.7. The chapter finishes with a discussion including possible programming extensions.

### 6.2 Introduction

Chapters 4 and 5 discussed point and variance estimation in IP weighted survival models, respectively. This chapter focuses on the related software and, in particular, the software to accompany the proposed M-estimation variance estimator. Point

estimates from an IP weighted analysis on survival data can easily be obtained in **Stata** (and **R**). In **Stata**, this involves performing a logistic regression model (with **logit**), generating the weights based on the propensity score, declaring the data as weighted survival data (with **stset** and **pweights**) and fitting a survival regression model (with **streg** or **stpm2** [42]). This analysis will provide robust standard errors. The process can easily be bootstrapped to obtain bootstrap standard errors. However, there is currently no command that provides a closed-form variance estimator in **Stata** for this method and no available command in either **R** or **Stata** that will easily provide M-estimation variances for IP weighted parametric survival models.

First, relevant **R** packages are evaluated. As discussed in Section 5.3.4, three closed-form variance estimators have been proposed for IP weighted survival data. Both Hajage *et al* [2] and Shu *et al* [21] provided accompanying **R** packages: **hrIPW** and **ipwCoxCSV**, respectively. The **R** code for Mao *et al*'s [20] estimator is available upon request from the author.

Point and variance estimates from any set of unbiased estimating equations can be obtained using the **geex** package in **R** [164]. The user must first define the set of estimating equations to be solved and then translate them into an **R** function [164]. In the case of IP weighted survival models, this will require the score function from the (logistic) treatment model and weighted survival outcome model. The parameter and empirical sandwich variance estimates are then calculated using numerical routines (for root solving and differentiation). Therefore, **geex** can be used to obtain the M-estimation variance estimator for IP weighted parametric survival models. However, the user is required to know and translate the score function and the numerical routines are more computationally intensive than using analytical formulas of the empirical sandwich variance estimator would be.

The main **Stata** command to calculate marginal effects is **teffects**. **teffects** calculates the average treatment effect, average treatment effect in the treated and potential-outcome means using a number of estimators including IP weighting. M-estimation is used for the variance estimator. The command treats the unobserved counterfactuals as a missing data problem. In the case of survival data, censoring

provides a second source of missing data. **stteffects**, the extension of **teffects** to survival data, implements censoring weights to address this. This approach has the main limitation of essentially extrapolating through up-weighting survival times to provide a difference in marginal mean survival time. As discussed in Section 2.2.3, RMST is preferred to MST to avoid extrapolation. In addition, the marginal estimands estimated by **stteffects** are restrictive and **stteffects** does not incorporate **stpm2** (flexible Royston-Parmar) models.

The aim of this chapter is to develop a new **Stata** command, **stipw**, to facilitate IP weighted analyses on survival data and to provide the corresponding M-estimation variance estimator. **stipw** version 1.0.0, dated 17.01.2022, is publicly available on the **SSC** archive and from GitHub at <https://github.com/Micki-Hill/stipw>. The code for **stipw** is included in Appendix C.

The remainder of the chapter is organised as follows: Section 6.3 describes the algorithm employed when **stipw** is used. Section 6.4 discusses the syntax and example code using **stipw** to perform the illustrative analyses in Section 5.7 is given in Section 6.5. The chapter finishes by suggesting programming extensions in the discussion.

## 6.3 **stipw** Algorithm

As with all survival commands in **Stata**, the data first needs to be **stset**. This specifies the survival time and event indicator variables. The user will then call **stipw**, specifying the treatment/exposure variable, the confounder variables to be adjusted for and the parametric survival model to be fitted. Other options are also available, see Section 6.4. **stipw** works in the following way:

1. First, **stipw** identifies the observations to be used in the analysis. Any observation with missing treatment/exposure values, missing confounder values and/or where **\_st** equals zero are excluded from all of the analysis. The latter is determined by **stset** and can occur, for example, if the event time is missing or if observations end on or before the time origin. The exclusion en-

sures the same data is used to fit the treatment/exposure model (modelling treatment/exposure against the confounders) and the weighted survival model (modelling survival against treatment/exposure). The variable `_stipw_flag` is created to indicate which observations are used in the analysis.

2. Logistic regression is used for the treatment/exposure model. The treatment/exposure variable is modelled incorporating the confounders. Options such as offset variables and no constant term are permitted.
3. If stabilised weights are required, a second logistic model is used to model the treatment/exposure variable with no covariates. This is performed to obtain the unconditional probability of treatment/exposure, used in the numerator of the stabilised weights. A model is used to obtain this value, rather than obtaining it empirically, so that this additional uncertainty can be incorporated in the M-estimation variance estimates.
4. The (first) treatment/exposure model is used to obtain the propensity score: the probability of receiving the treatment/experiencing the exposure, given the confounder values. `stipw` then creates the weights, either stabilised (the default, using the probability of treatment/exposure from the second logistic model as well) or unstabilised, and stores this in a newly created variable `_stipw_weight`. See Equations 2.23 and 2.24 for how the weights are calculated.
5. The data is then reset with the weights. This is done using `streset` with `pweights`. The original `stset` is preserved, so that the stored characteristics of the data (viewed with `char list`) are unchanged following the analysis with `stipw`. If this was not done, for example, the data would be set without weights before the analysis, but would be set with weights after the analysis. If unchanged, subsequent analyses performed would then be on the weighted dataset. Wherever possible, `stipw` avoids changing the original dataset and characteristics.

6. The specified survival model is then fit to the weighted data. The following parametric models are available for use with `stipw`: (`streg` models) exponential, Weibull, Gompertz, log-logistic, log-normal and (`stpm2` models) Royston-Parmar models on the hazard scale.
7. **Mata** (Stata's matrix language) is then used to obtain the M-estimation variance estimator. This follows the methods described in Section 5.4.4. Some brief programming details are given here:
  - (a) First, matrix  $\mathbf{A}$  from Equation 5.3 is estimated. This is calculated in sections as described in Section 5.4.4. Matrix  $\mathbf{A}_1$  (both weights) and scalar  $A_4$  (stabilised weights) are independent of the outcome survival model and are calculated following Equations 5.4 and 5.6, respectively.
  - (b) Matrices  $\mathbf{A}_2$  (unstabilised weights) and  $\mathbf{A}_5$  (stabilised weights) and vector  $\mathbf{A}_6$  (stabilised weights) are estimated following Equations 5.5, 5.7 and 5.8, respectively. They require the score function of the outcome survival model. This is calculated analytically for the specified survival model and can allow for left-truncated data and for treatment to be modelled as part of the ancillary parameter (where appropriate).
  - (c) Matrix  $\mathbf{A}_3$  (both weights) is then obtained. As mentioned in Section 5.4.4, this is the negative Hessian matrix. It can be obtained by multiplying the naive variance estimator provided during Step #6 by  $n$  and then inverting it. This was not possible for the Weibull model because, at the time of writing `stipw`, the model-based/naïve variance estimator stored for reference when the robust variance estimator is requested was incorrect. Instead, this is calculated analytically using Equation G.1 for the Weibull model with unstabilised weights (similar for stabilised weights).
  - (d) The sections of matrix  $\mathbf{A}$  are then combined and the matrix is inverted.
  - (e) Matrix  $\mathbf{B}$  and then  $\mathbf{V}$  are estimated following Equation 5.3.
8. The stored model variance estimates (from `streg` or `stpm2`) are updated with

the M-estimation variance estimates. The Wald test is also updated.

9. Lastly, the original `stset` is restored.

In order to use `stipw`, the data must be `stset` without weights and the data must be single-record-per-subject survival data. The treatment/exposure variable must be binary and fixed (does not vary over time). The confounders must be continuous or binary (categorical variables need to be made into multiple dummy variables) and fixed.

There are some restrictions to the survival models permitted. Only `stpm2` models on the hazard scale are permitted. Generalised gamma models, frailty and shared-frailty models, relative survival models and cure models are all not currently permitted and would constitute areas of future work.

The variables `_stipw_flag` and `_stipw_weight` are replaced, without warning, in subsequent runs of `stipw`.

Following `stipw`, standard survival postestimation commands can be used, for example, `predict` and `stcurve`. These use the updated variance estimates in the stored results.

## 6.4 `stipw` Syntax and Options

### 6.4.1 Syntax

`stipw` could have been expressed in a number of ways. The two main possibilities were: have `stipw` as a postestimation command to adjust the stored variance estimates or have `stipw` as a modelling command, including the treatment model in brackets and defining the outcome model with options of the main command. The former possibility was rejected in favour of the latter, as there would have been too many opportunities for user error in the former. For `stipw` to have been a postestimation function, this would have essentially required the user to perform Steps 1-6 in Section 6.3 themselves, storing the results of the treatment model to be fed into `stipw`. This would rely on the user correctly following each step, for example,

performing the whole analysis on the relevant non-missing subset of the data and correctly calculating the weights to use in `stset` with the `pweights` option. Due to the additional amount of user input required, it was also suspected that this approach would not be used as much in practice. Therefore, the second option was chosen.

The syntax for `stipw` is:

```
stipw (logit tvar tmvarlist [, tmoptions]) [if] [in] , distribution(distname)  
[options]
```

*tvar* is the treatment/exposure variable with values 1 denoting treatment/exposure and 0 denoting untreated/unexposed. *tmvarlist* is the list of confounders that are to be adjusted for with at least one variable specified. *tmoptions* specifies the options relating to the treatment/exposure model. The `distribution` option must be specified, possible options for *distname* are listed below. *options* lists all other possible options. The rest of this section describes the *tmoptions* and *options*. Most of this information can be found in the help file for `stipw`. Information specific to `stpm2` models used the `stpm2` help file [42]. The algorithm in Section 6.3 is referenced using “Step”.

### 6.4.2 *tmoptions*: Treatment/Exposure Model

These options correspond to the (first) logistic regression used to model the treatment/exposure (Step #2).

- **noconstant**: This option suppresses the constant term from the treatment/exposure model.
- **offset(*varname*)**: This option includes variable *varname* in the treatment/exposure model with the coefficient constrained to 1.
- **tcoef**: This option displays the coefficient table from the treatment/exposure model. The default is to omit the table.

### 6.4.3 *tmoptions* & *options*: Maximisation

- *maximize\_options*: These include the standard **Stata** maximisation options to control the maximisation process. When used as part of *tmoptions*, this corresponds to the (first) logistic regression used to model the treatment/exposure (Step #2). When used as part of *options*, this corresponds to the outcome model (Step #6).
- **lininit**: This option can only be specified for Royston-Parmar models (**distribution(rp)**). During the model fitting in Step #6, the initial values for **stpm2** are obtained by fitting only the first spline basis function (a linear function of log survival time).

### 6.4.4 *options*: Outcome Model - **streg**

These options correspond to the outcome model, fitted to the weighted survival data using **streg** (Step #6).

- **distribution(exponential|weibull|gompertz|loglogistic|lognormal)**:  
This option specifies the distribution to be fitted to the weighted survival data.  
This option is required (with one of these distributions or **rp**).
- **ancillary**: This option specifies that the treatment/exposure variable should be used to model the ancillary parameter. By default, the ancillary parameter does not depend on the treatment/exposure variable.
- **ocoef**: This option specifies that the intermediate coefficient table from the outcome model (Step #6) should be displayed. This will show the robust variance estimates before M-estimation has been performed. The default is to omit the table.
- **oheader**: This option will display the header information from the intermediate outcome model (Step #6). This will show extra information based on the robust variance estimates before M-estimation has been performed. The default is to omit the header.

### 6.4.5 *options*: Outcome Model - `stpm2`

These options correspond to the outcome model, fitted to the weighted survival data using `stpm2` (Step #6).

- **`distribution(rp)`**: This option specifies that a flexible parametric (Royston-Parmar) model should be fitted to the weighted survival data. This option is required (with `rp` or one of the distributions in the `streg` section above).
- **`bknots(knotslist)`**: This option specifies the boundary knots for the baseline hazard function. `knotslist` is a two-element numeric list, where the knots are specified on the scale defined by `knyscale()`. The default knot locations are located at the minimum and maximum of the uncensored survival times.
- **`bknotstvc(knotslist)`**: This option specifies the boundary knots for the treatment/exposure variable if it is specified as time-dependent. The default values are the same as above. `dftvc()` or `knotstvc()` need to be specified to use this option.
- **`df(#)`**: This option specifies the degrees of freedom for the baseline hazard function (between 1 to 10). See the `stpm2` help file for the default knot locations.
- **`dftvc(#)`**: This option specifies that the treatment/exposure variable is time-dependent and this is the degrees of freedom for this variable. With 1 degree of freedom, a linear effect of log time is fitted. See the `stpm2` help file for the default knot locations.
- **`failconvlininit`**: This option specifies that the `lininit` option should automatically be tried if convergence fails.
- **`knots(numlist)`**: This option allows the user to specify the knot locations for the baseline hazard on the scale defined by `knyscale()`. Exactly one of this option and `df()` should be specified.

- **knotstvc(*numlist*)**: This option specifies that the treatment/exposure variable is time-dependent and this is the knot locations for this variable. At most, only one of this option and **dftvc()** can be specified.
- **knyscale(*scale*)**: This option allows the user to specify the time-scale for the knots. **time** denotes the original scale (default), **log** denotes the log time scale and **centile** denotes the centile positions in the distribution of the uncensored log survival times.
- **noorthog**: This option suppresses orthogonal transformation of spline variables.
- **ocoef** and **oheader**: Same as in the section above.

#### 6.4.6 *options*: Variance Estimation

- **vce(vcetype)**: This option specifies the type of variance estimator used in the weighted outcome model. The options are **mestimation** (the default) and **robust**. If **robust** is specified, Steps #7 and #8 are not needed - a robust variance estimator is calculated by default in Step #6.

#### 6.4.7 *options*: Advanced

- **ipwtype(*string*)**: This option specifies the type of IP weight to use in the analysis. The options are **stabilised** (the default) and **unstabilised**. **stabilised** weights require Step #3, whereas **unstabilised** weights do not.
- **genweight(*newvar*)**: This option allows the user to name the variable that is created by **stipw** to store the weights. If this option is not specified, the weights are stored in **\_stipw\_weight** (Step #4).
- **genflag(*newvar*)**: This option allows the user to name the variable that is created by **stipw** to indicate which observations were used in the analysis. If this option is not specified, the indicator is saved as **\_stipw\_flag** (Step #1).

- **stsetupdate**: If this option is not specified, the original **stset** programmed by the user before **stipw** is called is preserved. If this option is specified, the **stset** with weights, as performed in Step #5, is saved instead (and Step #9 is not performed). Note, the user will not be able to run subsequent calls of **stipw** if **stsetupdate** is specified, as the data will then already be **stset** with weights.

#### 6.4.8 *options*: Reporting

These options refer to the final output following Step #8, rather than the intermediate output from Step #6. Superscripts <sup>1</sup> and <sup>2</sup> denote options only applicable to **streg** and **stpm2** outcome models, respectively.

- **level(#)**: This option specifies the confidence level, as a percentage, for confidence intervals. The default is 95.
- **noheader**: This option suppresses the output header.
- **nohr<sup>1</sup>**: This option specifies that coefficients rather than exponentiated coefficients be displayed, that is, that coefficients rather than hazard ratios be displayed. This is applicable to **exponential**, **weibull** and **gompertz** models, which give hazard ratios by default.
- **tratio<sup>1</sup>**: This option specifies that exponentiated coefficients, which are interpreted as time ratios, be displayed. This option is applicable to: **loglogistic**, **lognormal**, **exponential** and **weibull** models (when the latter two are fit in the accelerated time metric). The first two are given as coefficients by default.
- **noshow<sup>1</sup>**: This option prevents **stipw** from showing the key **st** variables.
- **eform<sup>2</sup>**: This option reports the exponentiated coefficients of Royston-Parmar models. The default is to report the coefficients. This gives the hazard ratio of the treatment/exposure variable if it is not time-dependent.

- **alleq**<sup>2</sup>: This option reports all equations used by `m1`. The models are fitted by using various constraints for parameters associated with the derivatives of the spline functions. These are not shown by default.
- **keepcons**<sup>2</sup>: This option prevents the constraints imposed by `stpm2` on the derivatives of the spline function when fitting delayed entry models being dropped. By default, the constraints are dropped.
- **showcons**<sup>2</sup>: This option displays the constraints used by `stpm2` of the derivatives of the spline function and when fitting delayed entry models. These are not listed by default.
- *display\_options*: These include some of the standard **Stata** display options.

## 6.5 Example Code with `stipw`

### 6.5.1 STD Dataset

The STD dataset was introduced in Section 1.3.5. The point estimates were obtained in Section 4.3.3 and the corresponding variance estimates were obtained in Section 5.7.2. The STD dataset was imported into **Stata** and prior to the analysis, some minor formatting took place:

- The time-to-event variable `time` was changed into `years`.
- The exposure variable (`race`) was changed to be numeric with 1 = Black, 0 = White.
- All categorical variables with more than 2 levels were made into dummy variables. This included the variables for marital status (`marital`), initial infection (`iinfct`) and condom use (`condom`).
- As described in Section 4.3.3, the variable for the number of sexual partners in the last 30 days (`npartner`) was transformed into a categorical variable

with  $\geq 3$  being grouped into a single level. This was then made into dummy variables.

The original and formatted variables are described in Table H.1 in the Appendix. The time-to-event variable was `years` and the event indicator was `rinfct`. The exposure variable was `race`, coded as 1 = Black and 0 = White. There were 15 confounders (20 dummy variables) included in the analysis, see Table H.1. The following code describes how the M-estimation variance estimates for the marginal hazard ratio and the difference in marginal RMST at time 4 years was obtained for Section 5.7.2.

First, the `stipw` package needs to be downloaded from the SSC.

```
ssc install stipw
```

Next, the data is loaded and declared as survival data without weights using `stset`.

```
use std, replace  
stset years, failure(rinfct)
```

`stipw` is then used to perform an IP weighted analysis. Inside the brackets describes the exposure model. A logistic regression model is used to model the exposure variable incorporating the 20 confounder (dummy) variables. The option then specifies a Weibull model (`distribution(weibull)`). By default, `stipw` uses stabilised weights and presents the M-estimation variance estimates of the parameters. The results table will show the marginal hazard ratio.

```
stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 ///  
npartner2 npartner3 npartner4 os12m rs12m condom2 condom3 ///  
abdpain discharge dysuria itch lesion rash lymph) , ///  
distribution(weibull)
```

```

.stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 npartner2 npartner3 npartner4 ///
>          os12m rs12m condom2 condom3 abdpain discharge dysuria itch lesion rash lymph) , ///
>          distribution(weibull)
Fitting logistic regression to obtain denominator for weights

Iteration 0:  log likelihood = -557.99159
Iteration 1:  log likelihood = -458.65576
Iteration 2:  log likelihood = -456.05012
Iteration 3:  log likelihood = -456.03964
Iteration 4:  log likelihood = -456.03964
Fitting second logistic regression with no confounders to obtain numerator for stabilised weights

Iteration 0:  log likelihood = -557.99159
Iteration 1:  log likelihood = -557.99159

Fitting weighted survival model to obtain point estimates

      Failure _d: rinfct
      Analysis time _t: years
      Weight: [pweight=_stipw_weight]

Fitting constant-only model:
Iteration 0:  log pseudolikelihood = -1020.2289
Iteration 1:  log pseudolikelihood = -995.16677
Iteration 2:  log pseudolikelihood = -994.89655
Iteration 3:  log pseudolikelihood = -994.89651

Fitting full model:
Iteration 0:  log pseudolikelihood = -994.89651
Iteration 1:  log pseudolikelihood = -994.2985
Iteration 2:  log pseudolikelihood = -994.29759
Iteration 3:  log pseudolikelihood = -994.29759

Displaying weighted survival model with M-estimation standard errors

Weibull PH regression                               Number of obs =     877
Log pseudolikelihood = -994.29759                  Wald chi2(1)   =    0.84
                                                       Prob > chi2   =  0.3583


```

<i>t</i>	M-estimation					
	Haz. ratio	std. err.	<i>z</i>	<i>P&gt; z </i>	[95% conf. interval]	
<i>race</i>	1.140326	.1629964	0.92	0.358	.8617075	1.509032
_cons	.3955661	.050498	-7.26	0.000	.3080028	.5080231
/ln_p	-.3005435	.0479627	-6.27	0.000	-.3945486	-.2065384
<i>p</i>	.7404157	.0355123			.6739842	.813395
1/ <i>p</i>	1.350593	.064778			1.229415	1.483714

**Figure 6.1:** Output for `stipw` used on the STD dataset with stabilised weights for a Weibull model

A Royston-Parmar model with 1 degree of freedom and the `noorthog` option is equivalent to fitting a Weibull model with `streg`. This approach has the benefit that the postestimation command `predict` can calculate the RMST. The option `eform` is also used so that the results will give the marginal hazard ratio (rather than the

marginal log hazard ratio). The above code can be written as a Royston-Parmar model with the following:

```

stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 ///
npartner2 npartner3 npartner4 os12m rs12m condom2 condom3 ///
abdpain discharge dysuria itch lesion rash lymph) , ///
distribution(rp) df(1) noorthog eform

. stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 npartner2 npartner3 npartner4 ///
>      os12m rs12m condom2 condom3 abdpain discharge dysuria itch lesion rash lymph) , ///
>      distribution(rp) df(1) noorthog eform
Fitting logistic regression to obtain denominator for weights

Iteration 0:  log likelihood = -557.99159
Iteration 1:  log likelihood = -458.65576
Iteration 2:  log likelihood = -456.05012
Iteration 3:  log likelihood = -456.03964
Iteration 4:  log likelihood = -456.03964
Fitting second logistic regression with no confounders to obtain numerator for stabilised weights

Iteration 0:  log likelihood = -557.99159
Iteration 1:  log likelihood = -557.99159

Fitting weighted survival model to obtain point estimates

Iteration 0:  log pseudolikelihood = -994.58556
Iteration 1:  log pseudolikelihood = -994.2978
Iteration 2:  log pseudolikelihood = -994.29759
Iteration 3:  log pseudolikelihood = -994.29759

Displaying weighted survival model with M-estimation standard errors

Log pseudolikelihood = -994.29759                               Number of obs = 877



|       | M-estimation |           |       |       |                      |
|-------|--------------|-----------|-------|-------|----------------------|
|       | exp(b)       | std. err. | z     | P> z  | [95% conf. interval] |
| xb    |              |           |       |       |                      |
| race  | 1.140326     | .1629963  | 0.92  | 0.358 | .8617075 1.509032    |
| _rcs1 | 2.096807     | .0744625  | 20.85 | 0.000 | 1.955826 2.24795     |
| _cons | -3955661     | .050498   | -7.26 | 0.000 | .3080028 .5080231    |



Note: Estimates are transformed only in the first equation.


```

**Figure 6.2:** Output for `stipw` used on the STD dataset with stabilised weights for a Royston-Parmar model with 1 degree of freedom and the `noorthog` option (equivalent to a Weibull model fitted with `streg`)

`predictnl` is needed to obtain the difference in marginal RMST at time 4 and the corresponding standard error. This will use the M-estimation variance estimates of the model parameters via the delta method.

```

predictnl sdrmst = predict(rmst at(race 1) tmax(4)) - ///
predict(rmst at(race 0) tmax(4)) in 1, se(sdrmst_se)

```

The call to `stipw` is repeated, but this time the `ipwtype` option is used to specify that unstabilised weights should be used.

```

stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 ///
npartner2 npartner3 npartner4 os12m rs12m condom2 condom3 ///
abdpain discharge dysuria itch lesion rash lymph) , ///
distribution(rp) df(1) noorthog eform ipwtype(unstabilised)

. stipw (logit race marital2 marital3 age yschool iinfct2 iinfct3 npartner2 npartner3 npartner4 ///
>          os12m rs12m condom2 condom3 abdpain discharge dysuria itch lesion rash lymph) , ///
>          distribution(rp) df(1) noorthog eform ipwtype(unstabilised)
Fitting logistic regression to obtain denominator for weights

Iteration 0:  log likelihood = -557.99159
Iteration 1:  log likelihood = -458.65576
Iteration 2:  log likelihood = -456.05012
Iteration 3:  log likelihood = -456.03964
Iteration 4:  log likelihood = -456.03964

Fitting weighted survival model to obtain point estimates

Iteration 0:  log pseudolikelihood = -1934.1618
Iteration 1:  log pseudolikelihood = -1933.8508
Iteration 2:  log pseudolikelihood = -1933.8507

Displaying weighted survival model with M-estimation standard errors

Log pseudolikelihood = -1933.8507                               Number of obs = 877



|             | M-estimation    |           |       |       |                      |          |
|-------------|-----------------|-----------|-------|-------|----------------------|----------|
|             | exp(b)          | std. err. | z     | P> z  | [95% conf. interval] |          |
| <b>xb</b>   |                 |           |       |       |                      |          |
| <b>race</b> | <b>1.141556</b> | .1625726  | 0.93  | 0.353 | .8635255             | 1.509106 |
| _rcs1       | 2.072608        | .0778626  | 19.40 | 0.000 | 1.925483             | 2.230975 |
| _cons       | .3967264        | .0504128  | -7.28 | 0.000 | .3092624             | .5089265 |



Note: Estimates are transformed only in the first equation.


```

**Figure 6.3:** Output for `stipw` used on the STD dataset with unstabilised weights for a Royston-Parmar model with 1 degree of freedom and the `noorthog` option (equivalent to a Weibull model fitted with `streg`)

The difference in marginal RMST is obtained similarly to before.

```

predictnl udrmst = predict(rmst at(race 1) tmax(4)) - ///
predict(rmst at(race 0) tmax(4)) in 1, se(udrmst_se)

```

## 6.5.2 AIDS Clinical Trials Group Study 175 (ACTG175)

### Dataset

The ACTG175 dataset was introduced in Section 1.3.1 and analysed in Section 5.7.3.

The ACTG175 dataset was imported into **Stata** and prior to the analysis, some minor formatting took place:

- The time-to-event variable **days** was changed into **years**.
- A new treatment variable (**trt**) was created to compare combination therapy with monotherapy.
- The antiretroviral history stratification variable (**strat**) was made into two dummy variables, as it was a categorical variable with three levels.

As before, the variables for zidovudine use prior to treatment initiation (**zprior**) and antiretroviral history (**str2**) were not included, see Section 5.7.3. The original and formatted variables are described in Tables H.2 and H.3 in the Appendix.

The time-to-event variable was **years** and the event indicator was **cens**. The treatment variable was **trt**, coded as 1 = combination therapy and 0 = monotherapy. There were 15 confounders (16 dummy variables) included in the analysis, see Tables H.2 and H.3. The following code describes how the M-estimation variance estimate for the difference in marginal RMST at time 3 years was obtained for Section 5.7.3.

First, the data is loaded and declared as survival data without weights using **stset**.

```
use actg175, replace  
stset years, failure(cens)
```

**stipw** is then used to perform an IP weighted analysis. Inside the brackets describes the treatment model. A logistic regression model is used to model the treatment variable incorporating the 16 confounder (dummy) variables. The options then

specify that a Royston-Parmar model (`distribution(rp)`) with two degrees of freedom (`df(2)`) should be fit to the weighted survival data. The treatment/exposure variable (`trt`) should be time-dependent with 1 degree of freedom (`dftvc(1)`). By default, `stipw` uses stabilised weights and presents the M-estimation variance estimates of the parameters.

```

stipw (logit trt age wtkg hemo homo drugs karnof oprior z30 ///
preanti race gender strat2 strat3 symptom cd40 cd80) , ///
distribution(rp) df(2) dftvc(1)

. stipw (logit trt age wtkg hemo homo drugs karnof oprior z30 preanti race gender strat2 strat3 ///
>         symptom cd40 cd80) , ///
>         distribution(rp) df(2) dftvc(1)
Fitting logistic regression to obtain denominator for weights

Iteration 0: log likelihood = -1482.1254
Iteration 1: log likelihood = -1476.1071
Iteration 2: log likelihood = -1476.1068
Iteration 3: log likelihood = -1476.1068
Fitting second logistic regression with no confounders to obtain numerator for stabilised weights

Iteration 0: log likelihood = -1482.1254
Iteration 1: log likelihood = -1482.1254

Fitting weighted survival model to obtain point estimates

Iteration 0: log pseudolikelihood = -1496.0573
Iteration 1: log pseudolikelihood = -1483.6534
Iteration 2: log pseudolikelihood = -1483.2159
Iteration 3: log pseudolikelihood = -1483.215
Iteration 4: log pseudolikelihood = -1483.215

Displaying weighted survival model with M-estimation standard errors

Log pseudolikelihood = -1483.215                               Number of obs = 2,139


```

	M-estimation					
	Coefficient	std. err.	z	P> z	[95% conf. interval]	
<b>xb</b>						
<trt></trt>	-.4811321	.0891676	-5.40	0.000	-.6558974	-.3063667
_rcs1	.7787811	.0453561	17.17	0.000	.6898848	.8676774
_rcs2	.1147734	.0343191	3.34	0.001	.0475092	.1820376
_rcs_trt1	.1704792	.0682318	2.50	0.012	.0367473	.3042111
_cons	-1.392623	.0574479	-24.24	0.000	-1.505219	-1.280027

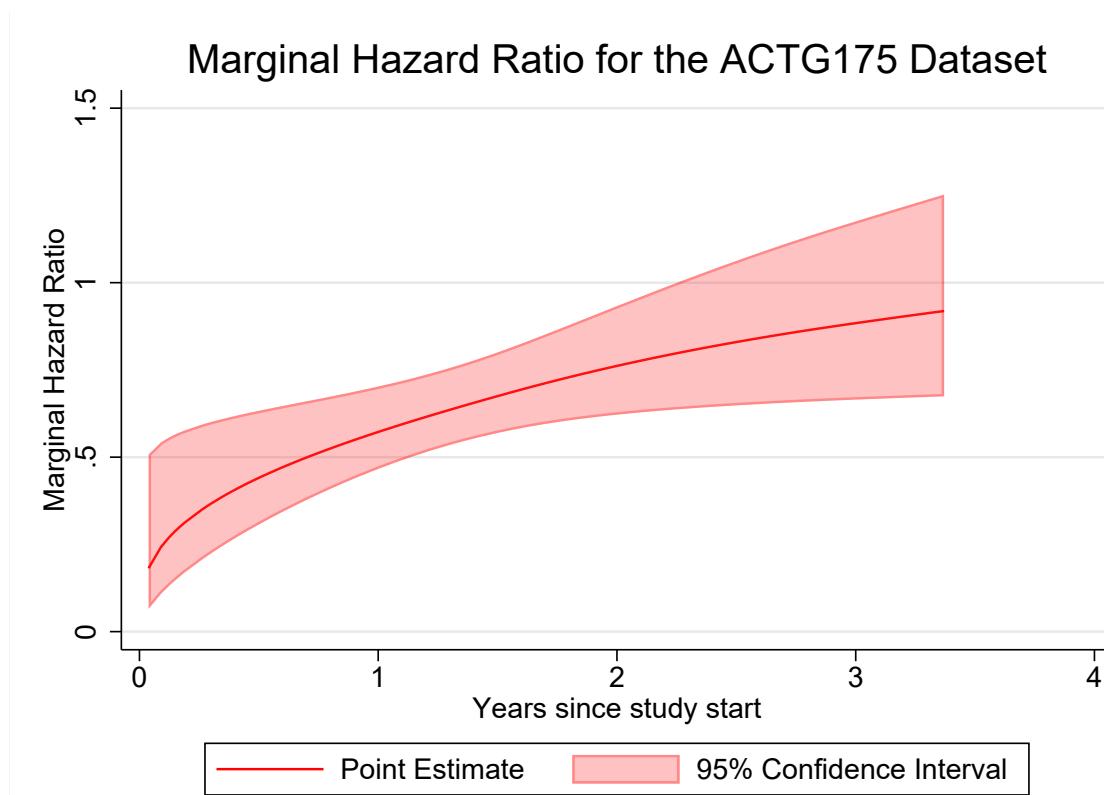
**Figure 6.4:** Output for `stipw` used on the ACTG175 dataset with stabilised weights for a Royston-Parmar model with 2 degrees of freedom where treatment is a time-dependent effect with 1 degree of freedom

In this example, treatment is modelled as a time-dependent effect and therefore the marginal hazard ratio cannot be summarised with a single value. Instead, the

marginal hazard ratio is a function of time. Another example of how postestimation easily follows `stipw` is that the marginal hazard ratio as a function of time can easily be calculated:

```
predict hr, hrnumerатор(trt 1) ci
```

The marginal hazard ratio is displayed with confidence intervals (based on the M-estimation variance estimates) in Figure 6.5.



**Figure 6.5:** The marginal hazard ratio for the ACTG175 dataset as a function of time with confidence intervals (based on the M-estimation variance estimates)

`predictnl` is needed to obtain the difference in marginal RMST at time 3 and the corresponding standard error.

```
predictnl sdrmst = predict(rmst at(trt 1) tmax(3)) - ///
predict(rmst at(trt 0) tmax(3)) in 1, se(sdrmst_se)
```

The call to `stipw` was repeated for unstabilised weights, similarly to Section 6.5.1.

### 6.5.3 Right Heart Catheterization (RHC) Dataset

The RHC dataset was introduced in Section 1.3.4 and analysed in Section 5.7.4. The RHC dataset was imported into **Stata** and brief details of the dataset will be given here. Prior to the analysis, some formatting took place:

- The event indicator **death** was changed to be numeric with 1 = died, 0 = censored.
- The time-to-event variable **years** was created and defined as years from study admission date (**sadmdte**) to death date (**dthdte**), if the individual died, or time from study admission date (**sadmdte**) to the maximum of the date of last contact (**lstctdte**) and hospital discharge date (**dschdte**), if the individual did not die.
- Categorical variables were changed to be numeric and transformed into dummy variables if there were more than 2 levels.
- As discussed in Section 5.7.4, lung and colon cancer were grouped in the primary and secondary disease category variables (**cat1** and **cat2**, respectively).

As before, the variables for urine output (**urine**), activities of daily life (**adld3pc**), weight (**wtkilo1**) and SUPPORT model estimate of the probability of surviving two months (**surv2md1**) were not included in the analysis, see Section 5.7.4. Observations with missing values (or illogical 0 entries) for covariates under the 5% threshold were dropped leaving 5509 individuals. This reduced dataset was renamed **rhc\_reduced**.

The time-to-event variable was **years** and the event indicator was **death**. The treatment variable was **rhc**, coded as 1 = RHC and 0 = no RHC. There were 49 confounders (67 dummy variables) included in the analysis, see Section 5.7.4. The following code describes how the M-estimation variance estimate for the difference in marginal RMST at time 5 years was obtained for Section 5.7.4.

First, the data is loaded and declared as survival data without weights using **stset**.

```
use rhc_reduced, replace
stset years, failure(death)
```

`stipw` is then used to perform an IP weighted analysis. Inside the brackets describes the treatment model. A logistic regression model is used to model the treatment variable incorporating the 67 confounder (dummy) variables (the start and end of the list of confounders are included in the code below). The options then specify that a Royston-Parmar model (`distribution(rp)`) with 5 degrees of freedom (`df(5)`) should be fit to the weighted survival data. The treatment variable (`rhc`) should be time-dependent with 2 degrees of freedom (`dftvc(2)`). By default, `stipw` uses stabilised weights and presents the M-estimation variance estimates of the parameters.

```
stipw (logit rhc age sex edu income1 income2 income3 ca_yes ca_meta ///
... ///
das2d3pc dnr1 aps1 scoma1 transhx) , ///
distribution(rp) df(5) dftvc(2)
```

`predictnl` is needed to obtain the difference in marginal RMST at time 5 and the corresponding standard error.

```
predictnl sdrmst = predict(rmst at(rhc 1) tmax(5)) - ///
predict(rmst at(rhc 0) tmax(5)) in 1, se(sdrmst_se)
```

The call to `stipw` was repeated for unstabilised weights, similarly to Section 6.5.1.

## 6.6 Discussion

Previously, it was only possible to obtain M-estimation variance estimates from an IP weighted parametric survival model by manually defining the estimating equations in `geex` in R or by using censoring weights and obtaining only a marginal MST using `stteffects` in Stata. A new Stata command, `stipw`, has been written as part of the work of this thesis to easily fit IP weighted parametric survival models and to provide the corresponding M-estimation variance estimator. This will be

quicker than `geex`, as analytical formulas of the variance estimator are given for standard parametric survival models (exponential, Weibull, Gompertz, log-logistic, log-normal and Royston-Parmar). It also does not require initial values for the parameters (these are calculated in the `stipw` algorithm using maximum likelihood). `stipw` is publicly available with an accompanying help file. It is easy to use and can perform IP weighted analyses with either stabilised or unstabilised weights in a few lines of code, as demonstrated in this chapter.

Many extensions could be added to `stipw` to improve its usability and breath. Firstly, most of the extensions mentioned in Section 5.8 apply here. These include incorporating different types of weights to target different estimands (for example, to target a marginal estimand in the treated); the extension to more than two levels of treatment and using a different model to estimate the propensity score (as long as it has a well-defined estimating equation). For example, `stteffects` allows for logistic, probit and heteroskedastic probit treatment/exposure models. Three further extensions, specific to `stipw`, are discussed below and include: checking the balance of confounders between treatment groups, IP weighted Cox models and a wrapper function for postestimation. Furthermore, the extension of `stipw` to generalised gamma models, frailty and shared-frailty models, relative survival models and cure models could be scope for future work.

The first main extension to `stipw` could be to include checks to ensure there is sufficient balance between the treatment groups in terms of the confounders after the IP weighting. After performing `stteffects` in Stata, the user can implement the command `tebalance` to summarise the standardised differences and variance ratio for the raw and weighted data (using `tebalance summarize`). An overidentification test for covariate balance can be performed using `tebalance overid` and density plots of the confounders from the raw and weighted data can be obtained using `tebalance density`. Two possible extensions to `stipw` could therefore be to: include the summary of the confounder balance between treatment groups and perform the overidentification test as part of the `stipw` algorithm. `stipw` could, for example, exit with an error if the treatment groups are insufficiently balanced after

weighting (violation of the conditional exchangeability assumption). The second option could be to provide a command similar to `tebalance`, which can be performed to check the balance after `stipw` has been executed. Currently, the propensity score model was assumed to have been correctly specified and the primary purpose of `stipw` was to provide M-estimation variance estimators. However, one of these two possible extensions could help to make `stipw` a more rounded tool for IP weighting analyses on survival data. A related, minor extension could be to include an option to store the estimates of the treatment model, so that the analyst could review this model after `stipw` has been performed.

This work has focused on parametric IP weighted survival models. One extension would be to provide the M-estimation variance estimator for an IP weighted Cox model (a model fitted with `stcox` in Stata). The theory has already been proposed by Shu *et al* [21], along with the R package `ipwCoxCSV`. Once the issue of the partial likelihood has been overcome, obtaining the M-estimation variance estimator for the marginal hazard ratio would be relatively straightforward and would correspond to including a single estimating equation for the outcome model. Obtaining the variance estimator for other marginal estimands, such as survival probabilities and RMST, would be more challenging as it would require estimating the baseline cumulative hazard. The uncertainty in this estimation would need to be taken into account by including it in the estimating equations. As far as it is known, this cannot be (easily) obtained in any software available currently.

Finally, many postestimation commands that follow `streg` and `stpm2` can be used after `stipw`. Examples include `stcurve` (following `streg`), `predict`, `predictnl` and `standsurv` [165]. This allows the user to estimate other useful marginal estimands, for example, marginal survival probabilities and RMST, with the corresponding variance estimates based on the M-estimation parameter variance estimates (via the delta method). A wrapper function specific to `stipw` could better facilitate postestimation, especially following `streg` models where the postestimation command `predict` does not include an option for RMST.

## 6.7 Conclusion

This chapter complimented Chapter 5 by discussing the accompanying software development. A new **Stata** command, `stipw`, has been written to easily fit IP weighted parametric survival models and provide the corresponding M-estimation variance estimator. It is publicly available and easy to use with either stabilised or unstabilised weights. Extensions could include adding tests to check the balance of the confounders between treatment groups, IP weighted Cox models and a wrapper function for postestimation.

# Chapter 7

---

## Multistate Model Application to the Hospital Acquired Infection Dataset and Corresponding Software Development

---

### 7.1 Outline

The aim of this chapter is to demonstrate how predictions can be obtained from a multistate survival model using a general simulation algorithm. These predictions are compared to those obtained from a simplified approach, which assumes constant transition rates and calculates the predictions analytically, and to non-parametric estimates. The chapter begins by explaining the methods for the application, including the general simulation algorithm. In order for non-parametric estimates to be obtained, the `msaj` command is majorly redeveloped and extended and the software development details are given next. The results of the analysis are then demonstrated and interpreted, before the chapter finishes with a discussion and conclusion.

### 7.2 Introduction

Multistate survival models were introduced in Section 1.1.4 and are being increasingly used to investigate complex disease pathways, for example, when interest lies in subsequent and/or intermediate events as well as a primary event. This unified

approach facilitates a better understanding of the whole disease profile and provides clinically relevant predictions, for example, transition probabilities and expected duration in each state. One example is in breast cancer, where the time to intermediate events, such as local recurrence and distant metastases, is of interest as well as overall survival [1]. Another example is repeated hospitalisations in patients with heart failure, where interest lies in the time spent in hospital (during each episode and in total) [166]. Further applications include other cancers (colorectal [167, 168], ovarian [169] and acute myeloid leukaemia [170]) and progression to diabetes [171].

Semi- and non-parametric methods have been frequently used to analyse multi-state models; however, interest is growing in parametric approaches. Although a variety of complex parametric models can relatively easily be fitted to each transition, see Section 2.7.2; the difficulty lies in obtaining the corresponding predictions from the full multistate model. A brief review of methods to obtain predictions was given in Section 2.7.3. In short, assuming an exponential or piecewise exponential Markov model allows for the transition probabilities to be calculated analytically. Alternatively, quadratic B-splines can be used to model the transitions in a Markov model and predictions can be obtained by numerically solving the forward Kolmogorov equations [106]. This chapter focuses on a general simulation algorithm to obtain predictions from a range of fitted parametric models [22], including Royston-Parmar models. In terms of implementation, available software includes: `mstate` in R [172] for semi- and non-parametric methods; `msm` in R [107] for exponential and piecewise exponential models; `flexsurv` in R [173] for fitting models and obtaining predictions by numerically solving the forward Kolmogorov equations; and `flexsurv` in R [173] or `multistate` in Stata [22] for the general simulation algorithm, the latter following model fitting by `merlin` [174, 175].

Von Cube *et al* [23] recommended an exponential multistate model as an accessible approach to obtain a quick, general understanding of the data. The authors demonstrated this method on HAI data: an extended illness-death model where a patient can have a HAI (intermediate event) and then/or be discharged or die (competing risks, the death/discharge with HAI are distinct from those without,

resulting in 6 states). This dataset was introduced in Section 1.3.3. Von Cube *et al* [23] acknowledged the potential implausibility of time constant transition rates (saying that this assumption is rarely met in practice) and recommended more sophisticated methods if the assumption was violated.

This chapter will illustrate a number of the methods described in Section 2.7 on the HAI dataset. In particular, the approach described in von Cube *et al* [23] is compared to the method described in Crowther and Lambert [22]. Importantly, time-dependent (and transition-specific) transition rates are supported with the latter approach. Another benefit of the latter approach is that functions of the transition probabilities (in this example, attributable mortality and population attributable fraction) can easily be obtained and, importantly, confidence intervals can be provided. The previous analysis by von Cube *et al* [23] is therefore extended by presenting confidence intervals and by estimating length of stay.

In order to compare the approaches of von Cube *et al* [23] and Crowther and Lambert [22], non-parametric Aalen-Johansen (AJ) estimates of the transition probabilities are needed. To facilitate this, the `msaj` command, part of the `multistate` package [22] in **Stata**, was majorly redeveloped and extended as part of the work of this thesis to provide a more comprehensive set of predictions including length of stay estimates. `msaj` version 1.0.1, dated 11.09.2020, is publicly available on the **SSC** archive through the `multistate` package and from GitHub at <https://github.com/RedDoorAnalytics/multistate/tree/main/msaj>. The code for `msaj` is included in Appendix D.

This chapter is organised as follows: Section 7.3 gives a more detailed description of the illustrative example, introduces the metrics of interest and explains the different analysis approaches. This includes giving details on the general simulation algorithm for obtaining predictions from a fitted multistate model in Section 7.3.4. The software development for `msaj` is discussed and demonstrated in Section 7.4 with example code. Section 7.5 displays the quantities of interest graphically, including comparisons between the approaches and confidence intervals for the predictions. The chapter finishes with a discussion and conclusion. This work has been published

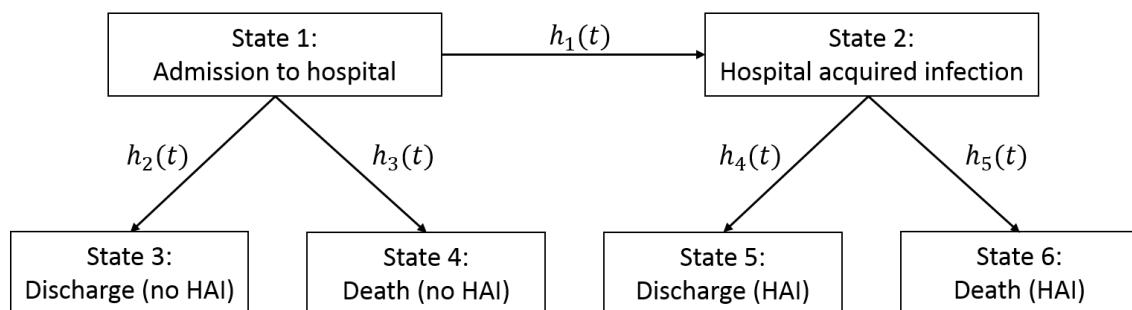
in BMC Medical Research Methodology [35]; see Appendix B.

## 7.3 Methods

### 7.3.1 The Extended Illness-death Model for HAIs

This chapter considers a multistate model in the context of HAIs, as previously described by von Cube *et al* [23]. When a patient is admitted to hospital, they are at risk of acquiring a HAI, which could lead to an increased hospital stay or increased risk of (hospital) death. An extended illness-death model with six states and five transitions, as illustrated in Figure 7.1, has been used to investigate the risks and consequences of HAIs. The time scale is days since hospital admission. All patients begin in state 1 at time 0, where the patient has been admitted to hospital but does not have an infection. The patient will then either become infected (state 2), be discharged without an infection (state 3) or die without an infection (state 4). If the patient acquires an infection, they will then either be discharged (state 5) or die (state 6) with an infection. The  $k^{th}$  transition rate from state  $a_k$  to  $b_k$  has been denoted as  $h_k(t)$  where:

$$\{a_1, a_2, a_3, a_4, a_5\} = \{1, 1, 1, 2, 2\} \quad \{b_1, b_2, b_3, b_4, b_5\} = \{2, 3, 4, 5, 6\}$$



**Figure 7.1:** Extended illness-death model for discharge and death with and without a hospital acquired infection (HAI)

### 7.3.2 Metrics of Interest

Recall the metrics defined for multistate models given in Section 2.7.1. These included the transition rates (Equation 2.29), transitions probabilities (Equation 2.27) and (possibly restricted and/or residual) expected length of stay (Equation 2.31). Note that the Markov assumption continues to be assumed in this chapter.

The first metric of interest was the transition probabilities from state 1 at time 0,  $P_{1b}(0, t)$ ,  $b = \{1, 2, 3, 4, 5, 6\}$ . By definition, HAIs take at least three days to develop [23] and so there are no HAI events prior to time 3 (3 days after hospital admission). Therefore, transition probabilities from state 2 at time 3 were also estimated,  $P_{2b}(3, t)$ ,  $b = \{2, 5, 6\}$ .

Following the formulas from Schumacher *et al* [176] and von Cube *et al* [23], let  $P_{1D}(s, t)$  denote the probability of dying (states 4 or 6) by time  $t$ , given a patient was in hospital without a HAI (state 1) at time  $s$ . Let  $P_{16|+}(s, t)$  denote the probability of dying with a HAI (state 6) by time  $t$ , given a patient was in hospital without a HAI (state 1) at time  $s$  and had become infected (states 2, 5 or 6) by time  $t$ . Similarly, let  $P_{14|-}(s, t)$  denote the probability of dying without a HAI (state 4) by time  $t$ , given a patient was in hospital without a HAI at time  $s$  and in a non-infected state (states 1, 3 or 4) at time  $t$ . The quantities can be calculated as follows:

$$P_{1D}(s, t) = P \{Y(t) \in \{4, 6\} | Y(s) = 1\} = P_{14}(s, t) + P_{16}(s, t)$$

$$\begin{aligned} P_{16|+}(s, t) &= P \{Y(t) = 6 | Y(s) = 1, Y(t) \in \{2, 5, 6\}\} \\ &= \frac{P_{16}(s, t)}{P_{12}(s, t) + P_{15}(s, t) + P_{16}(s, t)} \end{aligned}$$

$$\begin{aligned} P_{14|-}(s, t) &= P \{Y(t) = 4 | Y(s) = 1, Y(t) \in \{1, 3, 4\}\} \\ &= \frac{P_{14}(s, t)}{P_{11}(s, t) + P_{13}(s, t) + P_{14}(s, t)} \end{aligned}$$

The second set of metrics of interest were attributable mortality (AM) and pop-

ulation attributable fraction (PAF). AM and PAF can be used to investigate the excessive risk of dying due to HAIs [23, 176].

$$\text{AM}(s, t) = P_{16|+}(s, t) - P_{14|-}(s, t) \quad (7.1)$$

$$\text{PAF}(s, t) = \frac{P_{1D}(s, t) - P_{14|-}(s, t)}{P_{1D}(s, t)} \quad (7.2)$$

AM and PAF were estimated from time 0 (that is  $s = 0$  in Equations 7.1 and 7.2).

The third metric of interest was expected length of stay. Given a patient started in state 1 at time 0, the following quantities were estimated: restricted length of stay in state 1 ( $e_{11}(0, t)$ ), in state 2 ( $e_{12}(0, t)$ ) and overall in hospital ( $e_{11}(0, t) + e_{12}(0, t)$ ). Due to the three day delay in developing a HAI, the residual, restricted length of stay in state 2, conditional on having entered state 2 by time 3, was also of interest ( $e_{22}(3, t)$ ). These were calculated up until the last transition time (from any transition).

### 7.3.3 Analysis Approaches

The three (sets of) metrics were obtained from three models. The predictions from the different models were compared against each other and against non-parametric Aalen-Johansen estimates. The Aalen-Johansen estimator is discussed in detail in Section 7.4.

The first approach was to fit an exponential model to each of the transitions and obtain the metrics analytically, as was demonstrated by von Cube *et al* [23]. This approach was referred to as the “Exp” model.

The second approach was to select the best fitting distribution for each transition based on the AIC, henceforth denoted the “AIC” model. Following Crowther and Lambert [22], to each transition the following parametric models were applied: exponential, Weibull, Gompertz, log-logistic, log-normal, generalised gamma and Royston-Parmar models [8] with 2 to 5 degrees of freedom. The confidence intervals for the transition rates were obtained using the delta method. Once the multistate model was fitted analytically, with the best fitting distribution for each transition,

the general simulation algorithm was applied to obtain the metrics, see Section 7.3.4.

The third approach was to fit a Royston-Parmar model with 4 degrees of freedom to each of the transitions, henceforth denoted the “RP(4)” model. This was chosen as a reference parametric model for comparison purposes, as it should have sufficient flexibility to capture most complex shapes. As discussed in Section 2.3.4, three to five degrees of freedom should be sufficient to adequately capture the baseline hazard [43–45]. Once the multistate model was fitted analytically, the general simulation algorithm was used obtain the metrics.

### 7.3.4 Simulation Approach

The simulation algorithm works by projecting a patient through the multistate model in order to create their full event history. This is done a large number of times and the metrics of interest are calculated empirically from the large complete set of histories. The algorithm will now be described in brief, for further details see Crowther and Lambert [22].

A main component of this approach is to simulate event times from a fitted hazard function. This has already been performed for a Weibull model in the data generating mechanism sections in each simulation study, see Sections 3.5.2 and 4.5.2, and is described now in more detail. Consider the survival function given in Equation 2.1; it is bounded by the interval  $[0, 1]$ . Following Bender *et al* [177], let the survival function evaluated at simulated event time  $t$  equal  $u$ , a draw from the uniform distribution,  $\mathcal{U}(0, 1)$ .

$$S(t) = \exp \{-H(t)\} = u$$

Providing  $h(t) > 0$  and  $H(t)$  is an invertible function, this equation can be rearranged and directly solved for  $t$ .

$$t = H^{-1} \{-\log(u)\}$$

Crowther and Lambert [132] have extended the approach for when the event times follow a more complex distribution. They suggest numerical integration when  $H(t)$  does not have a closed form expression and iterative root finding to solve for  $t$  when  $H(t)$  is non-invertible.

The algorithm of Fiocco *et al* [99] and Crowther and Lambert [22] will be followed to obtain the transition probabilities from a multistate model. Let  $a$  be the starting state, entered at time  $t_a$ . If desired, specify a maximum follow-up time  $t_{max}$ . For ease of exposition, let the transition rates be represented as  $h_{ab}(t)$ , following Equation 2.28, rather than  $h_k(t)$ . For each simulated patient, repeat the following:

1. Let  $\mathcal{B}$  be the set of states that can be reached from state  $a$  and let  $N_a$  be the cardinality of set  $\mathcal{B}$ . If  $N_a = 0$  (that is,  $a$  is an absorbing state), stop. Otherwise, for each state  $b \in \mathcal{B}$ , let  $h_{ab}(t)$  represent the transition rate from  $a \rightarrow b$ .
2. For each state  $b \in \mathcal{B}$ , use  $h_{ab}(t)$  to simulate event times  $t_{ab}^*$  conditional on entering state  $a$  at time  $t_a$ . Event times are simulated using the general inversion method described above.
3. The event time is then  $t^* = \min(t_{a1}^*, \dots, t_{aN_a}^*, t_{max})$ . If  $t^* = t_{max}$ , stop.
4. Set  $a = c$  where  $t^* = t_{ac}^*$ ,  $c \in \mathcal{B}$  and set  $t_a = t^*$ .

The algorithm above is repeated for a large  $N$  number of patients. The transition probabilities are then estimated by calculating the proportion of people in each state at each time point of interest. The full event history of the simulated patients is known and therefore extended predictions can easily be obtained. For example, expected length of stay can be calculated by averaging the time spent in each state (up to each time point of interest) over all patients.

Let  $\widehat{\mathbf{b}}$  be the vector of parameter estimates and  $\widehat{\mathbf{V}}$  be the estimated variance-covariance matrix from the fitted multistate model (note that it is  $\widehat{\mathbf{b}}$  that is used to obtain the transition rates  $h_{ab}(t)$  in the algorithm above). Confidence intervals

can be obtained by drawing from a multivariate normal distribution with mean  $\hat{\mathbf{b}}$  and variance  $\hat{\mathbf{V}}$   $M$  times [22, 99]. For each draw  $m$ , the simulation algorithm above is repeated using the sampled  $\hat{\mathbf{b}}_m$  instead of  $\hat{\mathbf{b}}$  to calculate the transition rates (and therefore transition times). The variance of the  $M$  sets of estimates is then calculated and used to produce confidence intervals via normal approximation.

When using this approach, thought needs to be given on the choice of  $N$  (number of simulated patients) and  $M$  (number of draws from the multivariate distribution for the confidence intervals). The larger  $N$  and  $M$ , the smaller the MCSE; however, the greater the computation time. For this application,  $N = 1000000$  was chosen for point estimates and  $N = 100000$  and  $M = 500$  was chosen for the corresponding confidence intervals. To produce the confidence intervals for the transition probabilities from the “AIC” model, where probabilities were calculated at 165 equally spaced time points, took 48.5 minutes on a standard HP laptop with i5 processor and 8 GB of RAM.

### 7.3.5 Software

All analyses for this chapter were performed in **Stata** version 15.1. The parametric transition rates were obtained using **merlin**, version 1.12.0, dated 20/09/2020. All predictions and confidence intervals obtained via the general simulation algorithm were achieved using **predictms**, version 4.0.0, dated 28/10/2020, from the **multistate** package. The Aalen-Johansen estimates were obtained using **msaj**, version 1.0.1, dated 11/09/2020, also part of the **multistate** package. **msaj** was majorly redeveloped and extended and the software development is described in the next section.

## 7.4 Aalen-Johansen Estimates (Software Development)

### 7.4.1 Introduction

The Aalen-Johansen estimator provides non-parametric estimates for multistate Markov models and was introduced in Section 2.7.3. It was proposed by Aalen and Johansen [104] and has been discussed in detail by Andersen *et al* [103]. The Aalen-Johansen estimator has the advantage of making no assumptions on the shape of the transition rates and can provide summary metrics (for example, transition probabilities and expected length of stay) specific to the population sampled. It can also be a useful reference when comparing parametric multistate models, as will be demonstrated in Section 7.5. Parametric models may be needed, for example, for smoothed predictions or when extrapolation is necessary.

The Aalen-Johansen estimator has been implemented in a number of software packages, including **mstate** [172] and **etm** [31] in R and **multistate** [22] via the **msaj** command (version 0.6, dated 30.04.2019) in **Stata**. All three programs can calculate transition probabilities and Greenwood type standard errors. Both **mstate** and **etm** can produce expected length of stay estimates, allow for bi-directional models, allow for the entry time to be specified and give predictions from any starting state. **mstate** can also calculate Aalen standard errors, while the other two programs can be stratified by a covariate. Table 7.1 summarises the features available in each package/program.

The analysis of the HAI data motivated the development of **msaj** so that a comprehensive set of non-parametric estimates could be obtained. The first aim was to redevelop the structure of **msaj** to offer increased efficiency when obtaining point estimates and this was achieved by performing the processes in **Mata** rather than **Stata**. The second aim was to extend **msaj** to be able to output estimates from a user defined starting state and entry time, for bi-directional models and to calculate expected length of stay, similar to the functionality provided by **mstate**.

and `etm`.

**Table 7.1:** Summary of the key features of the `mstate` package (in R), `etm` package (in R) and `msaj` command (part of the `multistate` package in Stata) for calculating Aalen-Johansen estimates

Feature	<code>mstate</code>	<code>etm</code>	<code>msaj</code> Version 0.6	<code>msaj</code> Version 1.0.0
Transition probabilities	Yes	Yes	Yes	Yes
Standard errors	Greenwood	Greenwood	Greenwood	Greenwood
Aalen				
Expected length of stay	Yes	Yes	No	Yes
Can specify prediction time	Yes	Yes	No (Fixed as 0)	Yes
Prediction direction	Forward	Forward	Forward	Forward
Fixed horizon				
Can specify from state	Yes	Yes	No (Fixed as 1)	Yes
Can stratify by covariate	No	Yes	Yes	Yes
Can handle bidirectional models	Yes	Yes	No	Yes

This section begins by discussing the theory and implementation considerations for the Aalen-Johansen transition probabilities, standard errors and expected length of stay. It continues on to discuss other extensions included in the new `msaj` command (version 1.0.0, dated 02.03.2020), give example code corresponding to the predictions in Section 7.5 and notes possible future developments. Note that `msaj`, version 1.0.1, dated 11.09.2020, was specified in Chapter 7.3.5. The only difference from version 1.0.0 to version 1.0.1 is that the option `enter` was renamed to `ltruncate`.

### 7.4.2 Transition Probabilities

Following Allignol *et al* [31], let  $\mathbf{P}(s, t)$  be the  $N \times N$  matrix of transition probabilities  $P_{ab}(s, t)$ , where there are  $N$  states in the multistate model.  $\mathbf{P}(s, t)$  can be obtained from the transition rates through product integration [31]:

$$\mathbf{P}(s, t) = \prod_{(s,t]} \{\mathbf{I} + d\mathbf{H}(u)\} \quad (7.3)$$

Where  $\mathbf{I}$  is the  $N \times N$  identity matrix and  $\mathbf{H}(t)$  is the  $N \times N$  matrix of cumulative hazard functions  $H_{ab}(t)$ .

$\mathbf{H}(t)$  is estimated by the Nelson-Aalen estimator  $\widehat{\mathbf{H}}(t)$ . The elements of matrix  $\widehat{\mathbf{H}}(t)$  are obtained via:

$$\begin{aligned}\widehat{H}_{ab}(t) &= \int_0^t \frac{dN_{ab}(u)}{Y_a(u)} du \quad a \neq b \\ \widehat{H}_{aa}(t) &= - \sum_{b \neq a} \widehat{H}_{ab}(t)\end{aligned}$$

Where  $N_{ab}(t)$  is a counting process, calculating the number of direct transitions  $a \rightarrow b$  occurring up to time  $t$ .  $dN_{ab}(t)$  therefore represents the number of transitions from  $a \rightarrow b$  at time  $t$ .  $Y_a(t)$  is the number of patients in state  $a$  at the time immediately before  $t$ , that is, it is the number of patients at risk of the transition  $a \rightarrow b$  at time  $t$ .

By plugging the Nelson-Aalen estimator into Equation 7.3, the Aalen-Johansen estimator is obtained:

$$\widehat{\mathbf{P}}(s, t) = \prod_{(s,t]} \left\{ \mathbf{I} + d\widehat{\mathbf{H}}(u) \right\} \quad (7.4)$$

As  $\widehat{\mathbf{H}}(t)$  is a matrix of step functions, Equation 7.4 can be written as a finite matrix product, where the product is taken over all observed transition times  $t_k$  in  $(s, t]$  [31]:

$$\widehat{\mathbf{P}}(s, t) = \prod_{s < t_k \leq t} \left\{ \mathbf{I} + \Delta \widehat{\mathbf{H}}(t_k) \right\}$$

The Aalen-Johansen estimator is therefore a matrix of step functions, changing only at the times when a transition is observed. In `msaj`, transition probabilities are given at observed transition times only, until the last transition time (as the default).

### 7.4.3 Standard Errors

There are two methods for calculating standard errors for the Aalen-Johansen estimator: Aalen and Greenwood. `msaj` calculates Greenwood type standard er-

rors. Following Allignol *et al* [31] and Andersen *et al* [103], the Greenwood type covariance-variance matrix of the non-parametric transition matrix can be estimated by:

$$\widehat{\text{cov}} \left\{ \widehat{\mathbf{P}}(s, t) \right\} = \int_s^t \widehat{\mathbf{P}}(u, t)^\top \otimes \widehat{\mathbf{P}}(s, u-) \widehat{\text{cov}} \left\{ d\widehat{\mathbf{H}}(u) \right\} \widehat{\mathbf{P}}(u, t) \otimes \widehat{\mathbf{P}}(s, u-)^\top du \quad (7.5)$$

Where  $\otimes$  is the Kronecker product. If matrix  $\mathbf{A}$  has dimensions  $a_1 \times a_2$  and matrix  $\mathbf{B}$  has dimensions  $b_1 \times b_2$  then matrix  $\mathbf{A} \otimes \mathbf{B}$  will be a  $a_1 b_1 \times a_2 b_2$  matrix with each element in  $\mathbf{A}$  multiplied by each element in  $\mathbf{B}$ .

Computation of Equation 7.5 can be facilitated with the recursive formula:

$$\begin{aligned} \widehat{\text{cov}} \left\{ \widehat{\mathbf{P}}(s, t) \right\} &= \left[ \left\{ \mathbf{I} + \Delta \widehat{\mathbf{H}}(t) \right\}^\top \otimes \mathbf{I} \right] \widehat{\text{cov}} \left\{ \widehat{\mathbf{P}}(s, t-) \right\} \left[ \left\{ \mathbf{I} + \Delta \widehat{\mathbf{H}}(t) \right\} \otimes \mathbf{I} \right] + \\ &\quad \left\{ \mathbf{I} \otimes \widehat{\mathbf{P}}(s, t-) \right\} \widehat{\text{cov}} \left\{ \Delta \widehat{\mathbf{H}}(t) \right\} \left\{ \mathbf{I} \otimes \widehat{\mathbf{P}}(s, t-)^\top \right\} \end{aligned}$$

For a multistate model with  $N$  states,  $\mathbf{I}$  is the  $N \times N$  identity matrix.  $\widehat{\mathbf{P}}(s, t-)$  is the probability estimate from time  $s$  until a time just before  $t$ . As  $\widehat{\mathbf{P}}(s, t)$  is a step function,  $t-$  will correspond to the largest transition time smaller than  $t$ .  $\widehat{\text{cov}} \left\{ \Delta \widehat{\mathbf{H}}(t) \right\}$  is a  $N^2 \times N^2$  matrix and can be partitioned into blocks of  $N \times N$  matrices where only the diagonal elements are non-zero. The elements of  $\widehat{\text{cov}} \left\{ \Delta \widehat{\mathbf{H}}(t) \right\}$  are defined by:

$$\begin{aligned} \widehat{\text{cov}} \left\{ \Delta \widehat{H}_{ab}(t), \Delta \widehat{H}_{cd}(t) \right\} &= \\ \begin{cases} \{Y_a(t) - \Delta N_{a.}(t)\} \Delta N_{a.}(t) Y_a(t)^{-3}, & a = b = c = d \\ -\{Y_a(t) - \Delta N_{a.}(t)\} \Delta N_{ad} Y_a(t)^{-3}, & a = b = c \neq d \\ \{\delta_{bd} Y_a(t) - \Delta N_{ab}(t)\} \Delta N_{ad}(t) Y_a(t)^{-3}, & a = c, a \neq b, a \neq d \\ 0, & \text{Otherwise} \end{cases} & (7.6) \end{aligned}$$

$\Delta N_{a.}(t)$  is the number of transitions made from state  $a$  at time  $t$ .  $\Delta N_{ad}(t)$  represents the number of transitions made from  $a \rightarrow d$  at time  $t$ . The Kronecker delta function  $\delta_{bd}$  is an indicator function, equalling 1 if  $b = d$  and 0 otherwise. In

terms of the block notation,  $\widehat{\text{cov}} \left\{ \Delta \widehat{H}_{ab}(t), \Delta \widehat{H}_{cd}(t) \right\}$  can be found in the  $(b, d)^{th}$  block at the  $a^{th}$  row and  $c^{th}$  column. As covariance matrices are symmetrical, the second line in Equation 7.6 could more explicitly be defined as:

$$\begin{cases} -\{Y_a(t) - \Delta N_{a.}(t)\} \Delta N_{ad} Y_a(t)^{-3}, & a = b = c \neq d \\ -\{Y_a(t) - \Delta N_{a.}(t)\} \Delta N_{ab} Y_a(t)^{-3}, & a = c = d \neq b \end{cases}$$

`msaj` provides standard errors and confidence intervals (via normal approximation) for the Aalen-Johansen transition probabilities. Confidence intervals are truncated to ensure they are restricted to the  $[0, 1]$  interval. Confidence intervals could have been calculated on the  $\log(-\log(\cdot))$  scale and back-transformed to the original scale; however, following R packages `mstate` [172] and `etm` [31], the more simple approach was adopted.

#### 7.4.4 Length of Stay

The most notable extension to `msaj` was the addition of length of stay estimates. As the Aalen-Johansen estimator is a step function, expected length of can be calculated as a summation of rectangle areas. Formally, the expected (residual, restricted) length of stay,  $e_{ab}(s, t)$  from Equation 2.31, is estimated by the recursive formula:

$$\widehat{e}_{ab}(s, t) = \widehat{e}_{ab}(s, t_{-1}) + \widehat{P}_{ab}(s, t_{-1})(t - t_{-1})$$

$t_{-1}$  is the largest transition time (for any transition) where  $t_{-1} < t$ . If there are no transition times in the interval  $(s, t)$  then  $\widehat{e}_{aa}(s, t) = t - s$  and  $\widehat{e}_{ab}(s, t) = 0, a \neq b$ . Unlike the Aalen-Johansen estimator, the length of stay estimator is not a step function. `msaj`, version 1.0.0 onwards, calculates expected length of stay for all observed times (whether a transition occurred or was censored) up until the last transition time (as a default). This was to discourage extrapolation past the observed data. Currently, analytical standard errors are not supported and it is recommended that the user employs bootstrapping.

### 7.4.5 Other Extensions

Three further extensions are available in `msaj`, version 1.0.0 onwards:

- **Starting state not 1:** The program now allows the user to specify from which state they want predictions to be made. In other words, users can specify state  $a$  in the probability  $P_{ab}(s, t)$ , whereas before  $a = 1$ .
- **Non-zero entry time:** The program now allows the user to specify from what time they want predictions to be made. Exit times can also be specified. In other words, users can specify the times  $s$  and maximum  $t$  in the probability  $P_{ab}(s, t)$ , whereas before  $s = 0$  and maximum  $t = t_{max}$ , where  $t_{max}$  was the largest transition time.
- **Bidirectional models:** The program now allows for bidirectional models.

`msaj`, version 0.6, allowed estimates to be stratified by a covariate. This functionality remains in `msaj`, version 1.0.0 onwards, and is compatible with the extensions.

### 7.4.6 Example Code

Example code will now be given to obtain the non-parametric Aalen-Johansen estimates shown in Section 7.5. First, the data is loaded and the transition matrix is specified. This follows Figure 7.1 where the numbers in the matrix represent the transitions, the row represents the state being left and the column represents the state being entered.

```
ssc install multistate
use hai, replace
matrix tmat = (.,1,2,3,.,.\ .,.,.,.,4,5\ .,.,.,.,.\ .,.,.,.,.\ ///
.,.,.,.,.\ .,.,.,.,.)
```

The data is in long format where there is one row per individual per transition they are at risk of. The variable `_status` defines the transition indicator and `_start` and `_stop` give the start and stop times for when the individual is at risk of the

transition. Long format can be achieved using `msset`, see Crowther and Lambert [22]. The data is declared as survival data using `stset`.

```
stset _stop, failure(_status) enter(_start)
```

The data are now ready for `msaj`. The following is used to estimate the transition probabilities from state 1 at time 0 and corresponds to Figure 7.4.

```
msaj, transmatrix(tmat)
```

The following is used to estimate the transition probabilities from state 2 at time 3 and corresponds to Figure 7.5. This and the following calls to `msaj` demonstrate the new functionality in version 1.0.0 onwards.

```
msaj, transmatrix(tmat) from(2) ltruncated(3)
```

The following is used to estimate the corresponding standard errors and confidence intervals (estimates not shown).

```
msaj, transmatrix(tmat) from(2) ltruncated(3) ci se
```

The following is used to estimate the length of stay from state 1 at time 0 and corresponds to Figure 7.7.

```
msaj, transmatrix(tmat) los
```

The following is used to estimate the length of stay from state 2 at time 3 and corresponds to Figure 7.9.

```
msaj, transmatrix(tmat) los from(2) ltruncated(3)
```

#### 7.4.7 Discussion

`msaj`, version 1.0.1, dated 11.09.2020, is available on the SSC archive and the major redevelopments and extensions have been described in this section. The extensions have been summarised in Table 7.1 and included: calculating expected length of stay, allowing for bi-directional models and allowing the user to specify a starting

state and entry time. This work involved restructuring the original program, so that all calculations were performed in **Mata** and the program's computational efficiency was improved. Example code using **msaj**, version 1.0.1, was given in Section 7.4.6 and corresponds to the output in Section 7.5.

The purpose of this project was to allow **Stata** users to be able to calculate a comprehensive set of non-parametric estimates from a multistate model. Prior to the update, the range of estimates from **msaj** were limited and, as far as it was known, there were no other packages in **Stata** providing this functionality. **mstatecox** [178] is available in **Stata** for non-parametric and semi-parametric models. However, it calculates estimates using a general simulation algorithm, similar to **predictms** in **multistate** [22], rather than empirically.

**msaj** now has the same functionality as **etm**; however, cannot calculate fixed horizon predictions and Aalen type standard errors like **mstate**. One advantage of **msaj** is that length of stay estimates are provided for all observed times with one command call, which easily lends itself to displaying the results graphically. Alternatively, both the **R** packages would require repeated calls to their postestimation functions to plot the same graph. One feature not available in any package is (analytical) standard errors for expected length of stay. This could be a useful future extension and, in the interim, bootstrapping is recommended.

## 7.5 Results

### 7.5.1 Data

The HAI dataset was introduced in Section 1.3.3 and includes 756 patients admitted to hospital (all patients started in state 1 at time 0). 632 patients did not acquire an infection during the study, of which 475 were discharged and 157 died. 124 patients did acquire an infection, of which 90 were discharged and 34 died. There was no censoring in this sample and the last event occurred 82 days after admission.

### 7.5.2 Transition Rates

Table 7.2 gives the AIC for each distribution fitted to each transition separately. The AIC indicated that the following models gave the best fit for each transition and were therefore chosen for the “AIC” model:

1. **Transition 1:** Royston-Parmer model with 4 degrees of freedom.
2. **Transition 2:** Generalised gamma model.
3. **Transition 3:** Royston-Parmer model with 4 degrees of freedom.
4. **Transition 4:** Log-normal model.
5. **Transition 5:** Generalised gamma model.

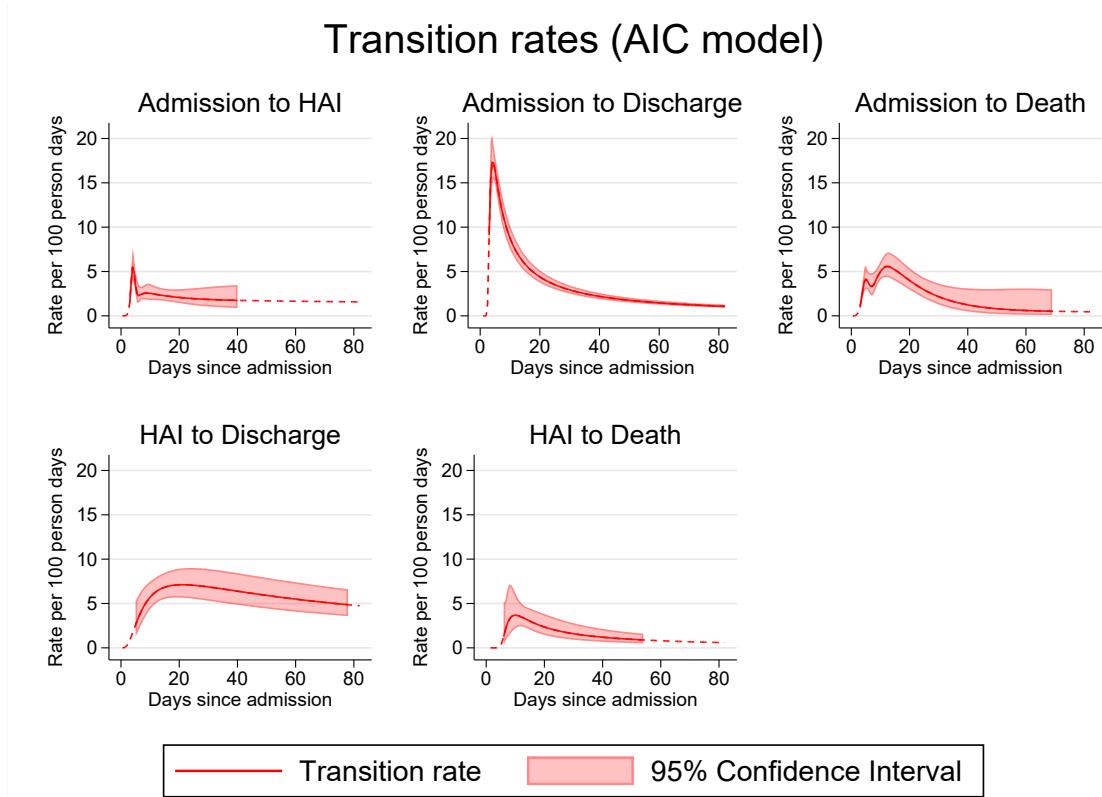
**Table 7.2:** AIC for each parametric model fitted to each transition separately (to determine the “AIC” model) on the HAI dataset

Model	Trans 1	Trans 2	Trans 3	Trans 4	Trans 5
Exponential	1229.7	3428.9	1482.3	691.6	328.7
Weibull	1208.6	3363.2	1425.9	692.5	330.0
Gompertz	1230.1	3430.1	1475.6	693.4	328.4
Log-logistic	1193.8	3204.2	1389.6	687.4	328.2
Log-normal	1175.7	3182.6	1374.7	<b>686.6</b>	328.0
Generalised gamma	1141.9	<b>3047.6</b>	1361.0	687.3	<b>325.0</b>
RP DF=2	1168.1	3163.1	1367.3	687.7	327.1
RP DF=3	1141.5	3081.4	1367.0	687.9	328.3
RP DF=4	<b>1136.8</b>	3072.5	<b>1361.0</b>	689.1	328.7
RP DF=5	1138.5	3070.3	1363.3	690.8	330.3

Trans = Transition, RP = Royston-Parmar, DF = Degrees of freedom

Figure 7.2 illustrates the transition rates from the “AIC” model. The point estimates and confidence intervals are shown from the time of the first event until the last event for each transition by a solid line. The corresponding intervals were [3, 40], [3, 82], [3, 69], [5, 78] and [6, 54] for transitions 1-5, respectively. The point estimates were extrapolated to cover the interval [0, 82] with a dashed line. It was evident that the transition rates were not constant over time and transition 2 (admission to discharge without HAI) appeared to deviate most drastically from

this assumption. Figure I.1 in the Appendix compares the transition rates from the “AIC” model to the “Exp” model, “RP(4)” model and to non-parametric estimates obtained using the Epanechnikov kernel smoother. The smoothed non-parametric estimates varied depending on the kernel type and bandwidth used; however, in all cases, the transition rates were clearly not constant.

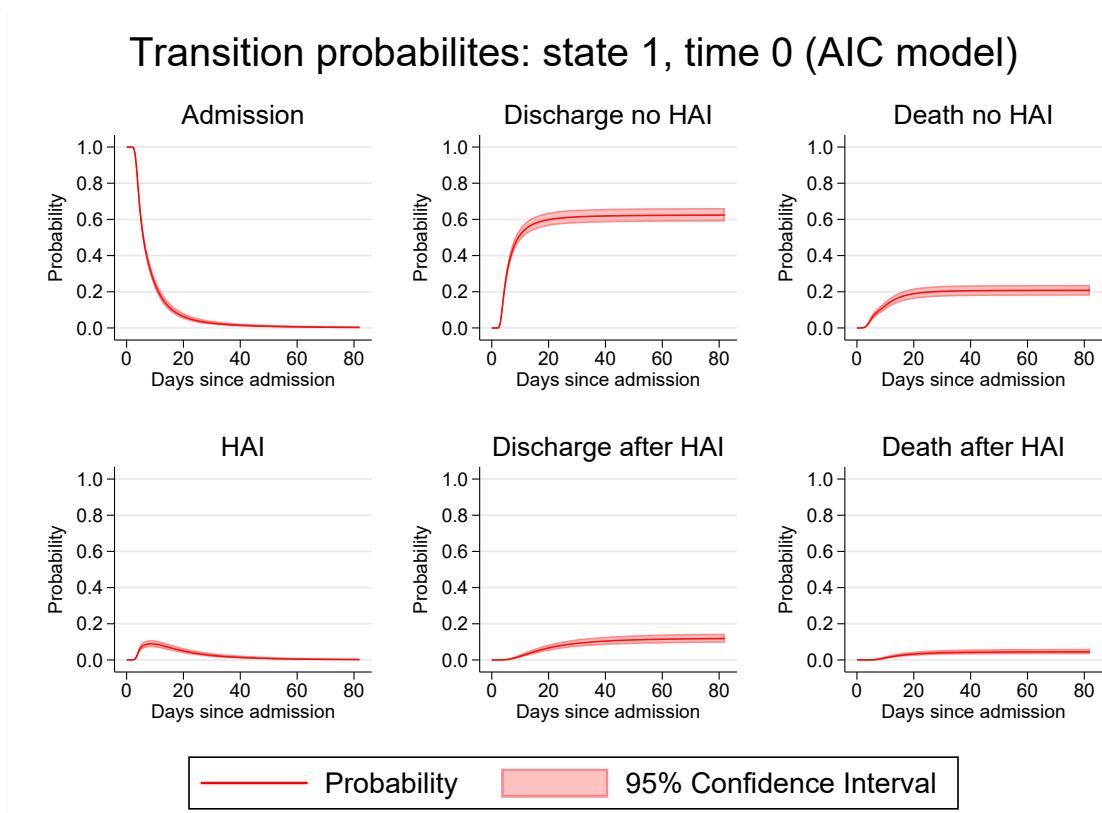


**Figure 7.2:** Transition rates from the “AIC” model with 95% confidence intervals (shaded region) for the HAI data. Point estimates and confidence intervals were defined from the time of the first event until the last event for each transition (solid lines). The point estimates were extrapolated to cover the interval [0, 82] (dashed line)

### 7.5.3 Transition Probabilities

Figure 7.3 gives the point estimates and 95% confidence intervals from the “AIC” model for the transition probabilities (starting in state 1 at time 0). The graphs can be interpreted as follows: given a patient was admitted to hospital (state 1) at time 0, the probability they were still in hospital without a HAI (state 1) 10 days later is 23.6% (95% CI: 20.7%, 26.1%). The probability at day 10 they were in hospital

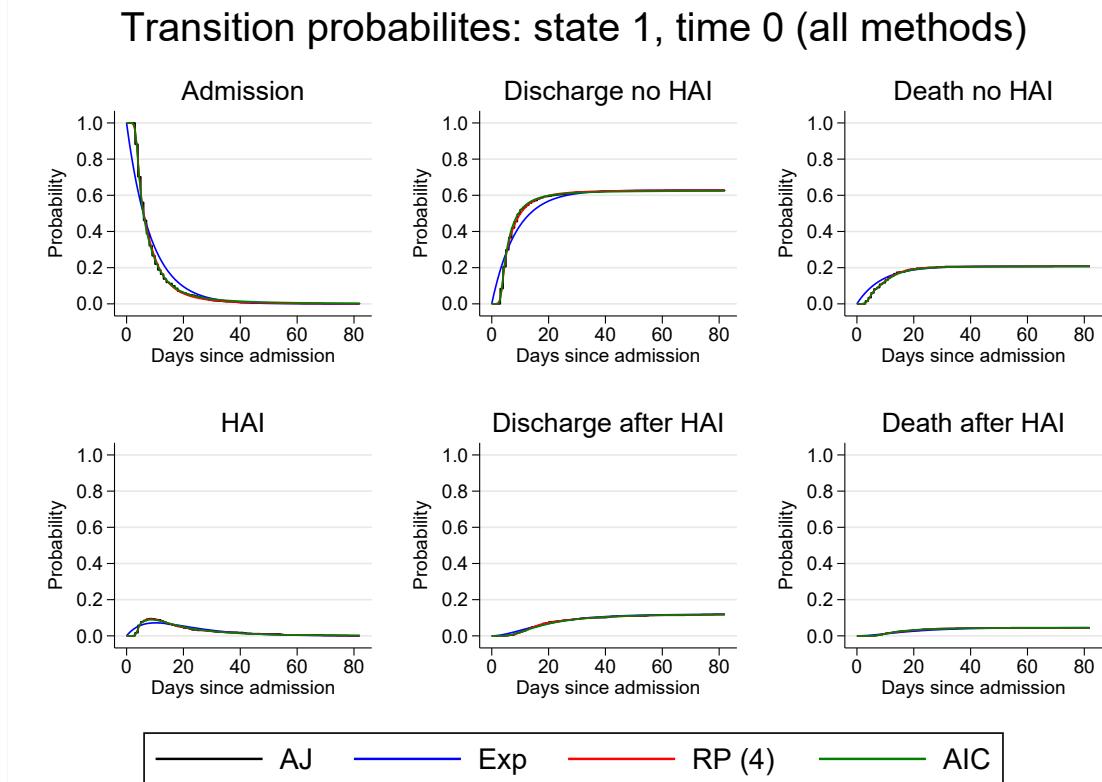
with a HAI (state 2) is 8.8% (95% CI: 6.8%, 10.4%). The probabilities at day 10 for the remaining states are 51.3%, 13.1%, 2.0% and 1.1%, respectively for discharge no HAI (state 3), death no HAI (state 4), discharge after HAI (state 5) and death after HAI (state 6).



**Figure 7.3:** Transition probabilities from state 1 at time 0 to each state for the “AIC” model with 95% confidence intervals for the HAI data

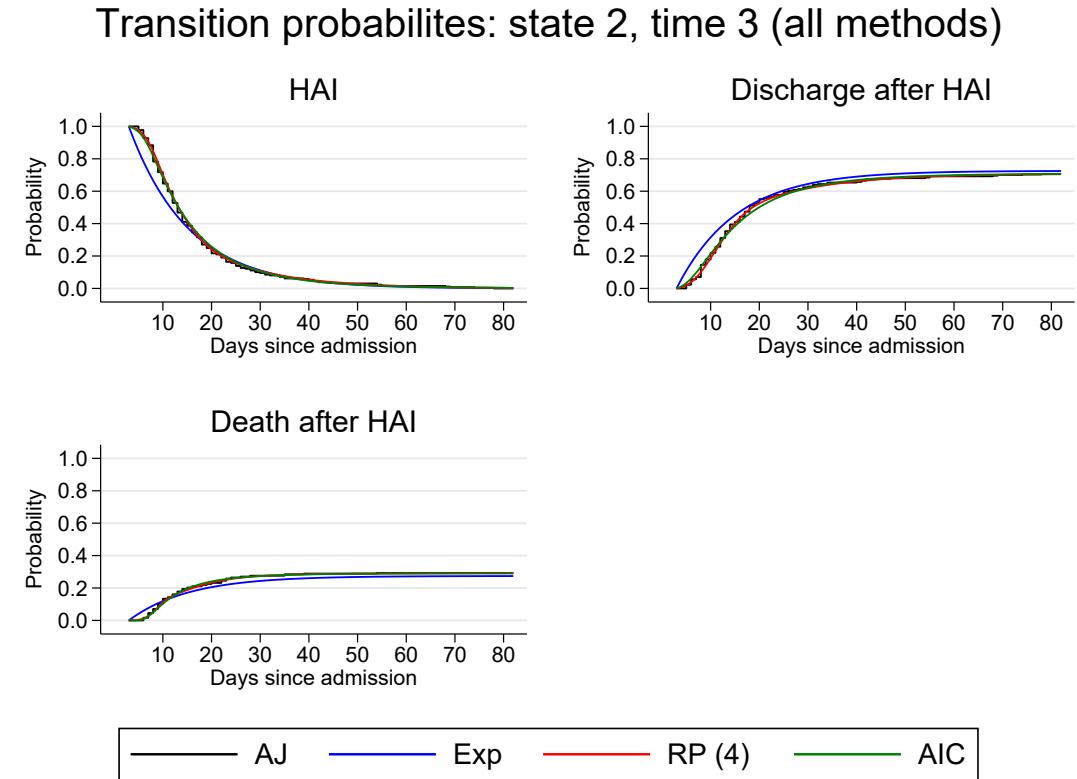
Figure 7.4 compares the transition probability estimates from the three approaches with non-parametric estimates (starting in state 1 at time 0). The predictions from the “AIC” and “RP(4)” models had high concordance with the Aalen-Johansen estimates. As von Cube *et al* [23] noted, despite clear departures from the constant hazard rates assumption, the “Exp” model performed well for states 4, 5 and 6 (death without HAI, discharge with HAI and death with HAI). There were some discrepancies with states 1, 2 and 3 (hospital admission without HAI, with HAI and discharge with HAI) up to 30 days after admission. Importantly, the predictions obtained from the “AIC” and “RP(4)” model captured the three day delay in acquiring a HAI (and in fact the delay in experiencing any event, as the

minimum event time was 3 days since admission), which the “Exp” model could not capture.



**Figure 7.4:** Transition probabilities from state 1 at time 0 to each state for the different approaches for the HAI data. Note that there is considerable overlap between the “RP(4)” and “AIC” estimates

Figure 7.5 illustrates the transition probabilities conditional on being in state 2 (in hospital with a HAI) by time 3. Predictions from the “AIC” and “RP(4)” models were slightly more consistent with the non-parametric estimates than those from the “Exp” model (especially in the first 20 days). Figure I.2 in the Appendix shows the corresponding 95% confidence intervals for the “AIC” model.

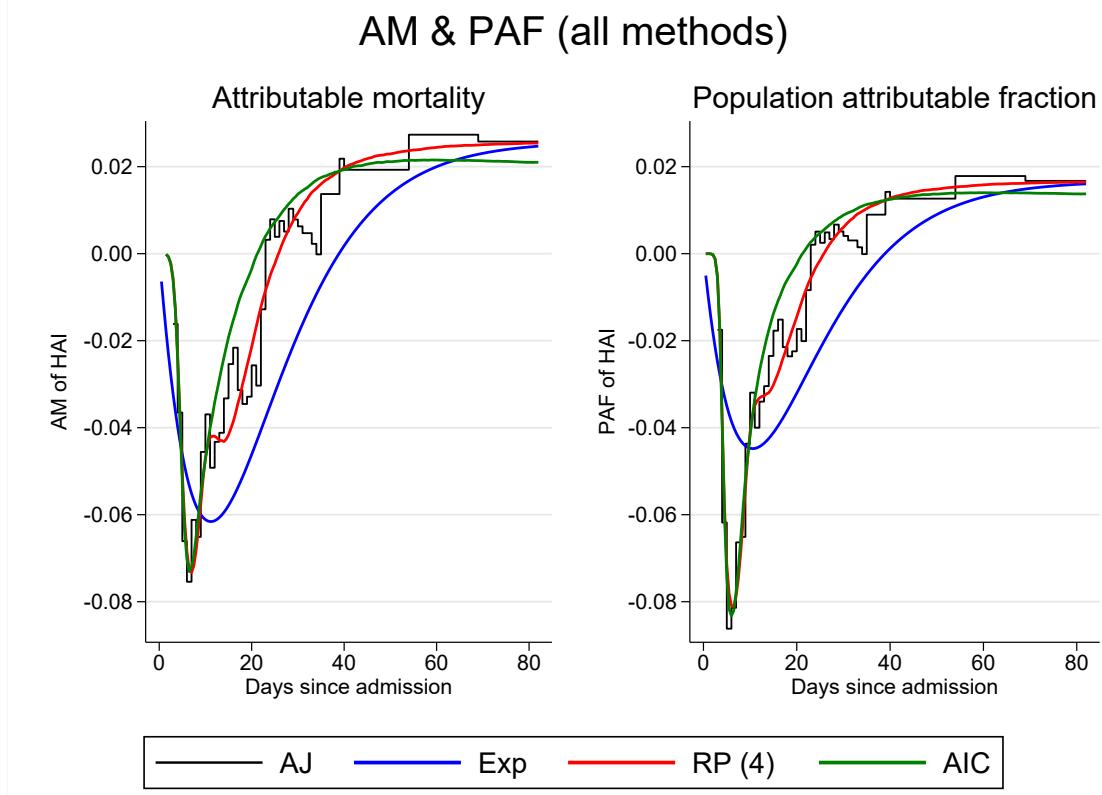


**Figure 7.5:** Transition probabilities from state 2 at time 3 to the relevant states for the different approaches for the HAI data

#### 7.5.4 Attributable Mortality and Population Attributable Fraction

About 20-25 days after hospital admission, AM was above 0 for the “AIC” and “RP(4)” models, see Figure 7.6. This suggested that after 25 days, the probability of dying was greater for those with an infection than for those without. AM can be interpreted as, for example: an individual that acquired a HAI by time 10 had a 4.7 percentage point decreased probability of dying by time 10 compared to an individual who did not acquire a HAI. Alternatively, an individual that acquired a HAI by time 30 had a 1.4 percentage point increased probability of dying by time 30. These results were similar for PAF, suggesting that after 25 days the occurrence of a HAI increased the risk of death and therefore the overall probability of dying. PAF can be interpreted as, for example: the proportion of individuals dying by time 10 would have increased by 4.2% if there were no HAIs. Alternatively, the proportion

of individuals dying by time 30 would have decreased by 0.9% if there were no HAIs (all predictions from the “AIC” model). Schumacher *et al* [176] describes the phenomenon of the AM and PAF initially being lower than 0.

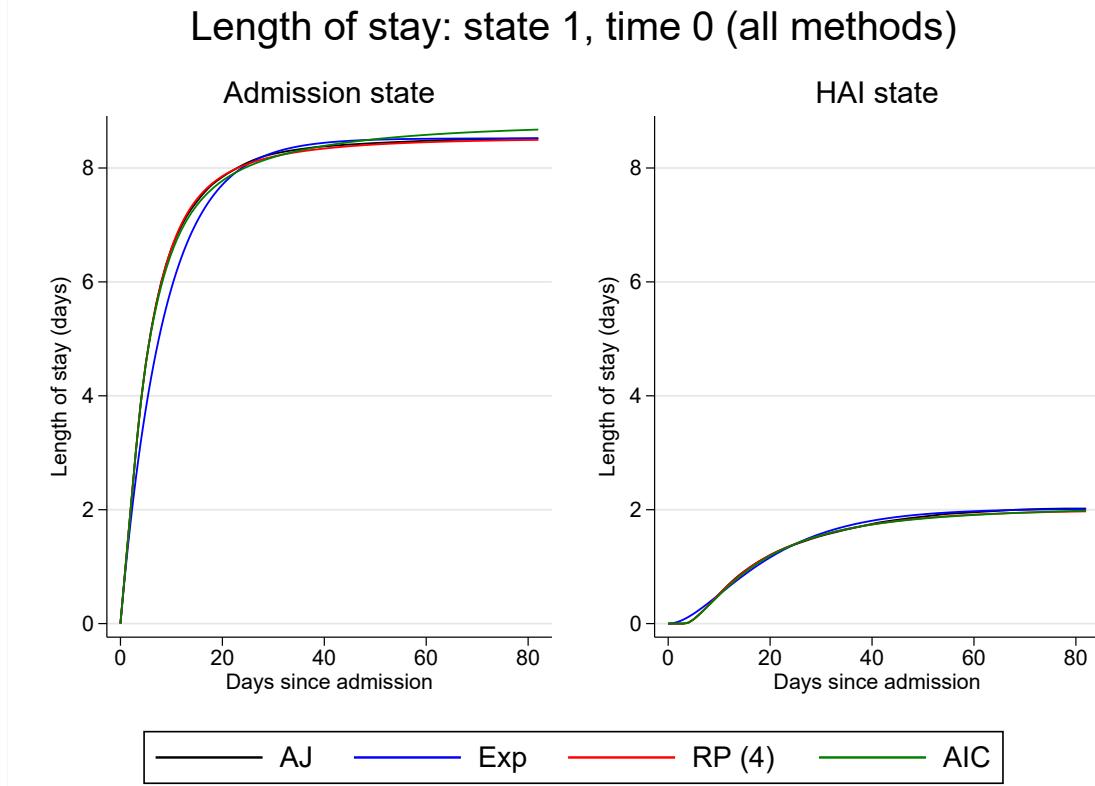


**Figure 7.6:** Attributable mortality (AM) and population attributable fraction (PAF) of HAIs for the different approaches for the HAI data

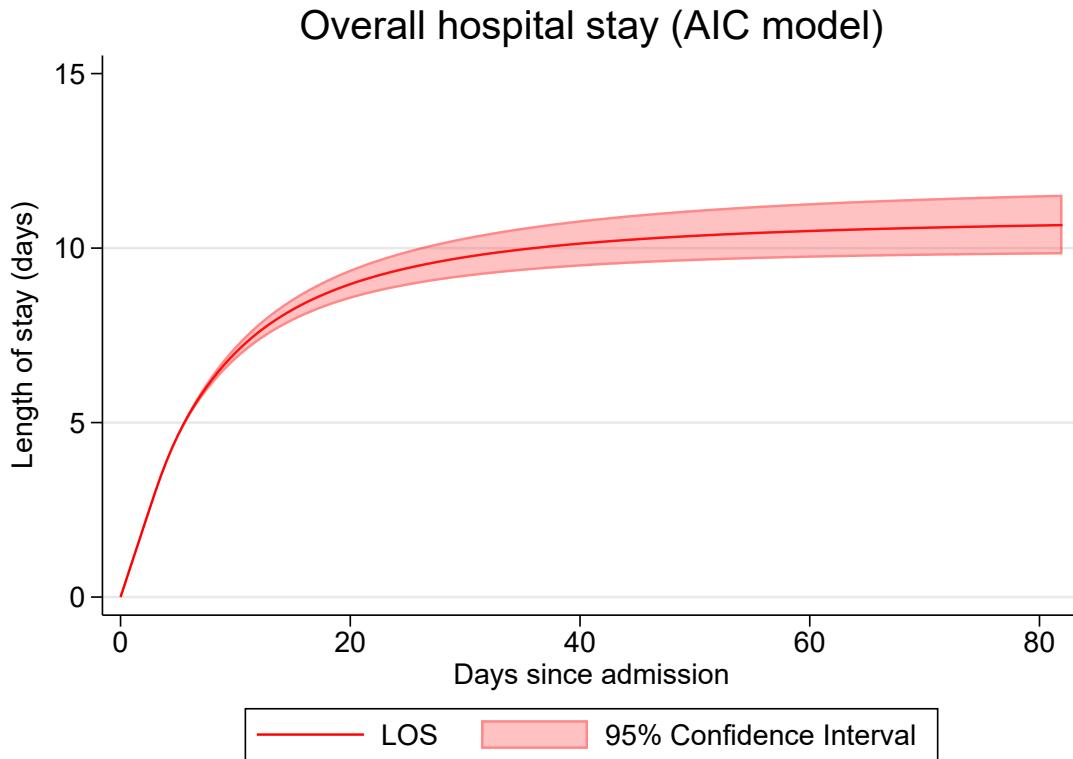
The relative differences in predictions between the models were greatest for AM and PAF (although the absolute differences were small). The “Exp” model appeared to overestimate (until 15 days after admission) and then underestimate AM and PAF considerably; whereas both the “AIC” and “RP(4)” predictions appeared to approximate the non-parametric estimates well. There was a slight inconsistency with the “AIC” model and the non-parametric estimates towards the end of the time window, although this should not be over-interrupted due to the small number of events occurring past 50 days. Figure I.3 in the Appendix shows the corresponding 95% confidence intervals for the “AIC” model.

### 7.5.5 Length of Stay

Figure 7.7 illustrates the restricted length of stay for the three models with non-parametric estimates (starting in state 1 at time 0). The graph can be interpreted as follows: 82 days since hospital admission, on average a patient spent 8.68 (95% CI: 8.04, 9.39) days in hospital without a HAI and 1.98 (95% CI: 1.53, 2.58) days in hospital with a HAI (estimates taken from the “AIC” model). The non-parametric restricted length of stay estimates were slightly better approximated by the “AIC” and “RP(4)” models (especially in the first 20 days). Figure I.4 in the Appendix shows the corresponding 95% confidence intervals for the “AIC” model and overall length of hospital stay with confidence intervals for the “AIC” model is shown in Figure 7.8.



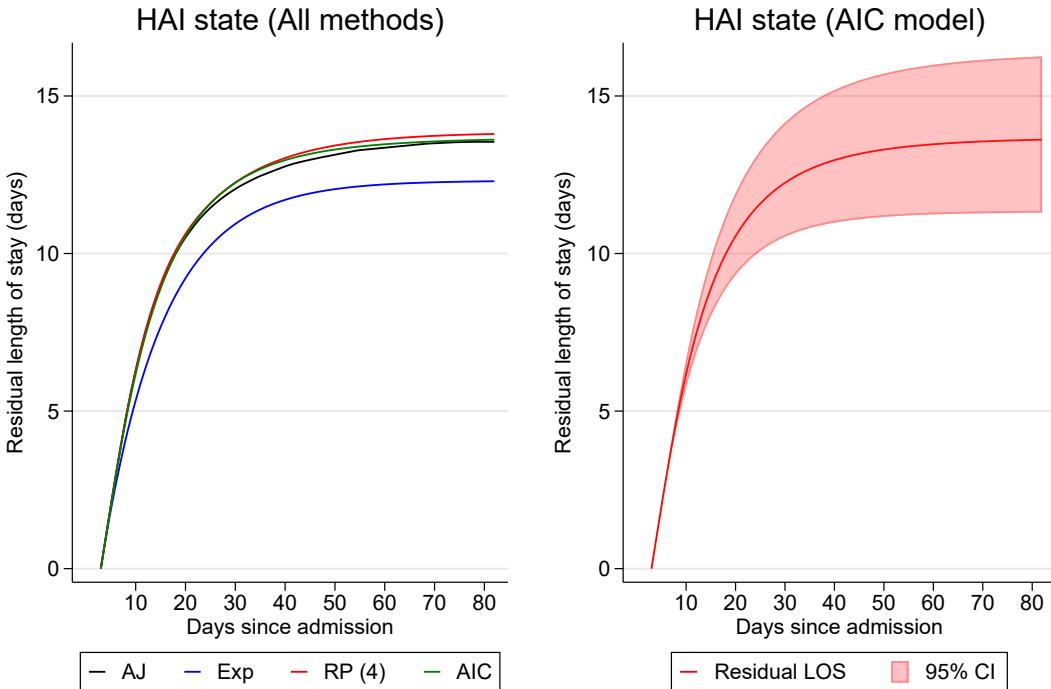
**Figure 7.7:** Length of stay in hospital without (state 1, left) and with (state 2, right) a HAI from state 1 at time 0 for the different approaches for the HAI data



**Figure 7.8:** Total length of stay in hospital from state 1 at time 0 for the “AIC” model with 95% confidence intervals for the HAI data

Figure 7.9 illustrates the residual, restricted expected length of stay, conditional on being in state 2 (in hospital with a HAI) by time 3. Between the interval [3, 82] days, a patient would have spent on average 13.61 (95% CI: 11.30, 16.26) days in hospital with a HAI, given they had a HAI and were in hospital by time 3 (estimates taken from the “AIC” model). While predictions obtained from the “AIC” and “RP(4)” models were consistent with non-parametric results, Figure 7.9 shows a large discrepancy between the latter and the “Exp” model. In the context of health economics, such differences could be non-trivial when translated into costs.

## Length of stay: state 2, time 3



**Figure 7.9:** Residual length of stay in hospital with a HAI (state 2), from state 2 at time 3 for the HAI data. In the left panel, for the different approaches. In the right panel, for the “AIC” model with 95% confidence intervals

## 7.6 Discussion

Assuming constant transition rates in a multistate model can facilitate a basic understanding of the data and this approach has been well demonstrated by von Cube *et al* [23]. However, this assumption may not always be plausible and, as a result, predictions may be misspecified. In the example shown, despite the transition rates not being constant (Section 7.5.2 and Figure 7.2), the transition probabilities from the “Exp” approach were similar to the Aalen-Johansen estimates (Section 7.5.3 and Figure 7.4). However, this was not the case for some functions of the transition probabilities, for example, AM and PAF (Section 7.5.4 and Figure 7.6). In addition, predictions from the “Exp” model starting in state 2 at time 3 had poorer concordance with the non-parametric estimates (Figures 7.5 and 7.9).

In this chapter, the “Exp” model was compared to two parametric models, where

predictions from the latter two fitted models were obtained via a general simulation algorithm, as described in Section 7.3.4. The “AIC” and “RP(4)” predictions were more consistent with the Aalen-Johansen estimates than the “Exp” model for all metrics. The greatest improvements were seen in AM and PAF and when considering delayed entry (predictions starting from state 2 at time 3).

As with any parametric approach, assumptions need to be made regarding the most appropriate distribution for each transition. A balance needs to be sought in terms of parsimony and sufficient parameters to appropriately capture the hazard shapes. This work has highlighted the challenges of model selection as the “AIC” model did not always have better concordance with the non-parametric estimates compared to the reference “RP(4)” model. Regardless of approach, sensitivity analyses around the assumptions of the baseline hazard are always recommended. It is important to note that both approaches still performed better than the “Exp” model. For this data example, a conditional parametric model would have been more appropriate for any transitions that could not have happened before day 3 (by definition or design of the study). A conditional model was not considered to be consistent with, and allow easier comparison with, the motivating paper by von Cube *et al* [23].

In addition to being able to model the transitions with a range of parametric distributions, the general simulation algorithm has other advantages. It easily lends itself to extended predictions, such as length of stay, the probability of ever visiting a state and disease specific quantities, such as AM and PAF. Uncertainties can also be easily obtained, which can facilitate a more comprehensive understanding of the data. There is also great flexibility available when modelling covariate effects, including time-dependent effects [22]. The approach generalises to more complex multistate models, for example, models with a greater number of states, greater number of transitions and backward transitions. It can also be applied to non-Markov models, unlike methods that rely on solving the forward Kolmogorov equations to obtain predictions.

A disadvantage of the general simulation algorithm is computational time. Al-

though point estimates can be obtained relatively quickly, confidence intervals can require a considerable amount of time, especially in the case of more complicated user-defined functions. A balance between computational time and MCSE is therefore needed when choosing  $N$  (number of simulations) and  $M$  (number of repetitions for confidence intervals). One possible alternative would be to use a hybrid approach when calculating predictions, where the transition rates obtained through parametric methods are substituted into the non-parametric Aalen-Johansen estimator [173, 179]. This approach would greatly decrease computational time; however, is only applicable to Markov models.

In this chapter, predictions have been obtained from a fitted multistate model using a general simulation algorithm, allowing for greater flexibility when modelling the transitions. This analysis can serve as a useful reference when the constant transition rates assumption is implausible and more complex transition rates are required. In this thesis, the command `msaj` was majorly redeveloped and extended, as described in Section 7.4, to provide non-parametric predictions for comparisons. A comprehensive set of predictions can now be obtained using `msaj`, version 1.0.0 onwards, including expected length of stay.

## 7.7 Conclusion

This chapter outlined the general simulation algorithm to obtaining transition probabilities and extended predictions from a fitted multistate model. This approach was demonstrated on an application to HAI data, following a recent publication [23], and the analysis can serve as a useful reference when the constant transition rates assumption is implausible. To aid comparisons, the **Stata** command `msaj`, part of the `multistate` package, was majorly redeveloped and extended to improve computational efficiency and to provide additional predictions such as expected length of stay.

# Chapter 8

---

## Discussion

---

### 8.1 Introduction

This chapter summarises the key findings of the thesis, relating back to the aims set in Section 1.2. Recommendations for future analyses are then given, along with a discussion of the potential impact this research can have in the related fields. The strengths and limitations of the projects are discussed and ideas relating to possible future work and developments are given. The chapter finishes with some concluding remarks and key messages.

### 8.2 Summary of Thesis

The overarching aim of the thesis was to develop and apply methods in parametric survival models to address, or investigate the impact of, issues that arise from both randomised clinical trials and observational data. Specifically, three topics in survival analysis were considered: interval censoring, IP weighting and multistate survival models. Chapter 1 introduced each topic and provided context and motivation for each of the four aims. The methods for standard survival analysis and each topic were then introduced in Chapter 2 and this provided a foundation for the subsequent chapters.

Chapter 3 addressed the first aim of investigating the performance of naive imputation techniques on interval-censored survival data. This was achieved by per-

forming a literature review of simulation studies and by completing a comprehensive simulation study to address the gaps in the literature. In particular, survival probabilities were investigated as an estimand, as they had not been explored in any prior simulation study. The first key finding of this work was that there were many scenarios where the naive methods performed poorly (biased point estimates, artificially precise standard errors and under-coverage, with the latter frequently occurring for the survival probability estimand) and scenarios where they performed reasonably well. Secondly, in general, midpoint imputation was the least biased naive method. Thirdly, increasing the interval width, decreasing the survival probability and increasing the treatment effect resulted in poorer performances for the naive methods. Increasing the sample size considerably reduced the coverage, while the impact of  $\gamma$  in the data generating mechanism varied across the imputation methods.

Chapter 4 addressed the second aim of confirming that both unstabilised and stabilised weights result in an unbiased estimator in an IP weighted survival analysis, with a fixed, binary treatment/exposure. A simulation studying considering fully, semi- and non-parametric methods was performed to confirm this expectation. The key finding of this work was that, in the majority of scenarios, stabilised and unstabilised weights performed similarly well in this setting and both provided generally unbiased estimates. The Kaplan-Meier estimator for marginal survival probabilities and RMST gave identical results for stabilised and unstabilised weights, as was proved in Section 4.4, and the exponential model gave near identical results, as both methods were saturated. In some scenarios, the Weibull model with stabilised weights gave very slightly less biased results than the corresponding unstabilised weights, although this was often of little practical importance. Alternatively, neither weight performed consistently better than the other for the Cox model.

Building on Chapter 4, the third aim of the thesis was to develop a closed-form variance estimator for IP weighted parametric survival models, which utilises M-estimation to take into account the associated uncertainty in the weight estimation. This aim was addressed in Chapters 5 and 6. Chapter 5 began by reviewing existing variance estimators before proposing the novel estimator for IP weighted parametric

models. The performance of the proposed estimator was evaluated in an extension of the simulation study performed in Chapter 4 and was demonstrated on three datasets. The key finding of this work was that, for large sample sizes, the proposed variance estimator performed similarly well to bootstrapping, giving relative percentage errors close to zero. By developing this variance estimator, the computational time required to obtain appropriate variance estimates for an IP weighted parametric model has been dramatically reduced and the results can be more easily reproduced. A second finding of this work was that the M-estimation variance estimator with stabilised and unstabilised weights gave similar estimates. Further work is needed to investigate the small sample properties of both the M-estimation and bootstrap variance estimators in this setting. A draft manuscript of this work is given in Appendix A.

Chapter 6 provided the accompanying software for the proposed variance estimator in Chapter 5. A new **Stata** command, **stipw**, has been written and made publicly available on the SSC archive and on Github at <https://github.com/Micki-Hill/stipw>, thus enabling other researchers to easily implement the methods. The command is provided in Appendix C. The algorithm and syntax were described in Chapter 6, along with example code for the illustrative analyses performed in Chapter 5.

Chapter 7 addressed the final aim of demonstrating how predictions can be obtained from a multistate survival model using the general simulation algorithm proposed by Crowther and Lambert [22]. This was achieved by extending a previous analysis [23], allowing for more complex transition rates, and by majorly redeveloping and extending the **Stata** command **msaj**, part of the **multistate** package [22]. The key findings from the analysis were that the more complex models (predictions obtained via the general simulation algorithm) gave predictions that were more consistent with the non-parametric estimates than the exponential model (predictions obtained analytically) for all metrics. This was especially clear for some functions of the transition probabilities and when starting in a later state at a later time. Another key result from this work was the development of **msaj** to pro-

vide a more comprehensive range of non-parametric predictions, including expected length of stay, as well as improved computational efficiency. This development was necessary to facilitate the comparisons between the parametric approaches. This work has been published in BMC Medical Research Methodology [35] and a copy of the manuscript is provided in Appendix B. The code for `msaj` is publicly available on the SSC archive, via the `multistate` package, and on GitHub at <https://github.com/RedDoorAnalytics/multistate/tree/main/msaj>. The command is provided in Appendix D.

## 8.3 Project Specific Recommendations and Potential Impact of Research

As expected, the recommendation when analysing interval-censored data is to use an appropriate method instead of single, naive imputation. With a number of tutorials [3, 67–69], appropriate methods [70, 74, 79, 126, 128] and software (`stintreg` and `stintcox` in **Stata** and various R packages [27, 29, 134, 135]) available for interval-censored data, there is little reason to use naive imputation, especially in light of its inferior performance in the simulation study. If the interval width is large, differs between groups [114] and/or is irregular [123], there is even more reason to avoid naive imputation. In addition, if the event times are thought to follow a Weibull distribution, a rapidly decreasing hazard would strongly motivate the use of appropriate methods. It is likely that this problem will arise for any distribution with a rapidly decreasing hazard function. If naive imputation must be used, midpoint imputation would be recommended in general. This result is intuitive, as midpoint imputation is the only naive technique that takes into account the length of the interval with the imputation.

Panageas *et al* [112] discussed the impact of the visit schedule on the median progression-free survival time when naive imputation methods were used. Despite this commentary and the published simulation studies identified in the literature review that highlight the limitations of naive imputation, end imputation is still

frequently used in clinical trials, see recent examples [115–118]. This work provides further reasoning as to why naive imputation should be avoided in favour of appropriate methods. In particular, this work has built on that by Panageas *et al* [112], by providing evidence for a measure of absolute risk, which, arguably, was more negatively affected by naive imputation than the hazard ratio in the simulation study.

The next recommendations are in the context of IP weighted analyses on survival data with a fixed, binary treatment/exposure variable. Firstly, either unstabilised or stabilised weights are recommended when the sample size is large. This holds for point and variance estimation (when an appropriate variance estimator is used). However, there may be a slight advantage in some scenarios to using stabilised weights when the treatment prevalence is small. In terms of variance estimation, the bootstrap variance estimator is one appropriate choice, as also recommended by Austin [19]. Similarly to Hajage *et al* [2], Mao *et al* [20] and Shu *et al* [21], this work also recommends a closed-form variance estimator as a possible alternative to bootstrapping in large sample sizes. The proposed M-estimation variance estimator may be preferred to bootstrapping, for example, if reproducibility or computational time are key concerns. Further work is required for recommendations regarding small samples.

This thesis has contributed to the IP weighting literature in three main ways. The first contribution is by confirming that either stabilised or unstabilised weights can be used in an IP weighted analysis on survival data with a fixed, binary treatment/exposure variable. The weighting strategies can result in slightly different results and the possible impact of the choice of weight had not previously been explored in this context in practice. The second contribution is by presenting a closed-form variance estimator for a range of parametric IP weighted survival models. Providing a computationally efficient and easily reproducible variance estimator can allow analysts to quickly perform multiple analyses, for example, for sensitivity analyses, while still making valid inferences based on correct variance estimates. Previous work had focused on directly modelling [21], linearising [2] or Poisson-

ising [20] the Cox model. By proposing a variance estimator for *parametric* IP weighted survival models, the variance of other predictions can easily be obtained. This facilitates the variance estimation of absolute risks, such as marginal survival probabilities and RMST, which can give a useful summary of the potential impact of the treatment/exposure. Finally, making newly developed methods accessible is crucial for their potential impact. Therefore, the third contribution of the thesis is providing the accompanying user-friendly software for the proposed M-estimation variance estimator.

The final recommendations refer to multistate survival models. The main recommendation when performing a multistate survival analysis is to model the transitions parametrically and obtain predictions via a general simulation algorithm [22]. Parametric models, namely Royston-Parmar models, can allow for complex hazard functions and the general simulation approach to obtaining predictions allows for great flexibility, for example, allowing for transition-specific distributions and sharing parameters across transitions. This approach requires a decision on  $N$ , the number of simulations, and  $M$ , the number of repetitions for the confidence intervals. In Chapter 7,  $N = 1000000$  was chosen for point estimates and  $N = 100000$  and  $M = 500$  were chosen for the corresponding confidence intervals. A balance between the MCSE and computational time needs to be sought with the choice of  $N$  and  $M$ . As with all analyses, it is also recommended to check any assumptions made, for example, the Markov assumption and the distribution of the transitions. This thesis has focused on Markov models; however, the general simulation algorithm can be used to obtain predictions when this assumption is relaxed. The model fit can be assessed by comparing the predictions against non-parametric estimates.

The two main contributions of this work to the multistate literature are demonstrating how predictions can be obtained using a recently proposed approach and developing and extending one of the accompanying software commands. The application serves as a comprehensive, worked example of the approach and software proposed by Crowther and Lambert [22]. Analysts wishing to perform their own multistate analysis can follow this implementation to gain a comprehensive under-

standing of their data (with uncertainties), without having to assume the transitions are constant. As mentioned above, providing accompanying software is vital for the accessibility of methods. The second contribution of this work to the field is majorly redeveloping and extending the command `msaj`. This allows users to perform a comprehensive non-parametric analysis as a preliminary, model checking or final analysis.

## 8.4 Strengths and General Recommendations

The first strength of the thesis is the celebration of parametric survival models. The proposed M-estimation variance estimator in Chapter 5 is a prime example of how the thesis has focused on methods for parametric models. Parametric models are often overlooked in favour of the semi-parametric Cox model or non-parametric estimators. This is because the latter require fewer assumptions, for example, the Cox model makes no assumptions on the underlying shape of the baseline hazard. However, with the introduction of flexible Royston-Parmar models, many complex shapes in the baseline hazard function can be captured. Royston-Parmar models also have the advantage of being able to easily accommodate time-dependent effects through interactions with spline terms. Parametric models more easily facilitate a range of metrics, such as survival probabilities and RMST, as the metrics are simply functions of the estimated parameters. The corresponding uncertainties are straightforward to calculate using the delta method. When it is sensible, parametric models also lend themselves more naturally to extrapolation. This may be needed, for example, in health economics where the effect of treatment on survival may be extrapolated to a longer period than the follow-up time, often to a lifetime horizon [180, 181]. A general recommendation from the thesis is to consider the use of parametric models when performing an analysis, especially if time-dependent effects, measures of absolute risk or extrapolation are required.

A second, related strength of the thesis is the focus on absolute risks, instead of solely the hazard ratio. Despite the hazard ratio having some well-known limita-

tions, as discussed in Section 2.6.2, it is still a commonly used estimand in survival analysis. One limitation of the hazard ratio is that the proportional hazards assumption is required to be valid. This assumption can be relaxed by expressing the hazard ratio as a function of time; however, it is then no longer a single summary measure. The survival probability and RMST have been used frequently in the thesis to summarise survival data. Although a time point is still required for these measures (normally the maximum follow-up time is a sensible suggestion for RMST), they may be more comparable across studies in similar individuals than the hazard ratio. This is because the hazard ratio is a conditional measure and will depend on what other covariates are included in the model. Absolute risks can also be compared across studies when different analysis models have been used and the assumption of proportional hazards is not required. As mentioned in Section 2.6.2, another drawback of the hazard ratio is that it is not collapsible, unlike the other estimands. This means that even in the absence of confounding, the conditional and marginal effects do not coincide [86]. A second, general recommendation is to consider absolute risks when performing an analysis, as they may more appropriately address the research question than relative risks or be a useful addition to give a more comprehensive summary of the survival data.

A further strength of the thesis is providing answers to research questions using simulation studies. For example, the simulation study in Chapter 3 addressed the gaps in the literature on the impact of naive imputation on interval-censored data. Likewise, the simulation study in Chapter 4 confirmed that either stabilised or unstabilised weights can be used in an IP weighted analysis on survival data. A simulation study was also used to evaluate the performance of the newly proposed variance estimator in Chapter 5.

Throughout the thesis, consideration was given on how best to communicate the methods and results of the simulation studies. The excellent tutorial by Morris *et al* [4] was followed using the ADEMP structure (aims, data generating mechanisms, estimands, methods and performance measures) and it is strongly recommended that analysts wishing to perform a simulation study follow this guide. Thought was

given as to the most appropriate performance measures for each aim and absolute risks were included in the estimands investigated, tying in with the previous strength of the thesis. Consideration was also given to the MCSE. Firstly, an iteration sample size calculation was performed for each study, so that the MCSE in the main study would be within an acceptable level. Where feasible, the MCSE (or corresponding confidence interval) was also shown in addition to the point estimates, either graphically or in tables, as would be expected when performing any analysis on real data.

A fourth strength of the thesis is the inclusion of motivating and illustrative datasets. In many cases, the datasets were used to motivate the research questions and demonstrate how different methods resulted in different estimates. For example, the breast cosmesis dataset was used to illustrate the impact of naive imputation on interval-censored data in Chapter 3, while the STD dataset was used to demonstrate how different weights gave different estimates in an IP weighted analysis in Chapter 4. In addition, the simulated data in Chapter 3 was reflective of real life as the parameters of the data generating mechanism were based on the breast cosmesis dataset. The datasets were also used to demonstrate relatively new methods and software. For example, the novel M-estimation variance estimator was illustrated on the ACTG175, STD and RHC datasets in Chapter 5, with the accompanying code given in Chapter 6. Likewise, the general simulation algorithm for obtaining predictions in multistate models was demonstrated in Chapter 7 on the HAI dataset.

Finally, the greatest strength of the thesis is providing user-friendly software to accompany the proposed methods. As discussed in Section 8.3, accessible software is paramount if methods are to be implemented in practice. A primary example of this is the development of **stipw**. **stipw** allows users to easily perform an IP weighted analysis on survival data using either stabilised or unstabilised weights. The proposed M-estimation variance estimator is used to estimate the variance of the parameters in the weighted outcome survival model. The code for **stipw** can be found in Appendix C and examples of **stipw** in action can be found in Chapter 6.

The other command showcased in the thesis was **msaj**, part of the **multistate**

package [22]. `msaj` can be used to obtain Aalen-Johansen estimates for a multi-state survival model. This command was majorly redeveloped and extended for the calculation of expected length of stay and to provide more flexibility with the predictions offered. A discussion of `msaj` is given in Section 7.4, with example code given in Section 7.4.6, and the command code is provided in Appendix D. The final recommendation is to provide at least example code, or preferably user-friendly software, for any newly proposed methods, so that they may be implemented by other analysts.

## 8.5 Limitations

As with any simulation study, a number of extensions could have been considered. Firstly, in all simulation studies, a Weibull data generating mechanism was used. Instead, a range of distributions could have been investigated or a more complex model, for example fractional polynomials, mixture model or Royston-Parmar model, could have been used to simulate more biologically plausible data. However, the former two models would have led to questions regarding model selection and model misspecification, as a different analysis method to the data generating model would be used. Secondly, the number of covariates could have been varied and time-dependent effects could have been included. For the simulation studies on IP weighting, the strength of the confounder variables could also have been varied. For these simulation studies, it could have been useful to explore the robustness of the estimators (point and variance) to departures of the conditional exchangeability and positivity assumptions as well as model misspecification.

A further limitation of the simulation studies in Chapters 4 and 5 was the lack of direct effect from covariates to outcome in the data generating mechanism. While the employed approach allowed for data to be generated from a marginal outcome model, therefore avoiding model misspecification, it did not follow the more natural, sequential data generating mechanism often found in causal inference simulations. As the sequential mechanism would generate data from a conditional proportional

hazards model, the resulting marginal model would be of a more complex form [152]. Therefore, the natural sequential mechanism was not used in favour of the marginal data generating mechanism and the limitations of this approach were acknowledged.

A restricted investigation into small sample properties, design of the study when small sample sizes were included and the handling of unconverged or extreme results also limited the simulation studies. While the simulation study on interval-censored data did consider small samples ( $n_{obs} = 100, 500$ ), the simulation studies on IP weighting focused on large sample properties ( $n_{obs} = 2000, 10000$ ). A small sample size of  $n_{obs} = 200$  was included in the evaluation of the proposed variance estimator in Chapter 5; however, the design of the simulation study was not necessarily appropriate. For example, difficulties arose for the bootstrap variance estimator when few individuals were assigned to the treatment group (non-convergence in a small proportion of cases) and when there were few events in one of the treatment groups (high standard errors for the marginal log hazard ratio). These were due to one or more extreme bootstrap samples. There were also incidences of considerably large (absolute) marginal log hazard ratios, suggesting non-convergence, when there were no events in one of the treatment groups. This affected all methods and if analysing a similar, single dataset outside of a simulation study, alternate analyses would be employed. Issues also arose for the smaller sample size in the interval-censored data simulation study. The difficulties occurred for the appropriate likelihood-based method when there were only left- and right-censored events in both treatment groups (non-convergence) or only left-censored events in the treated group (high standard errors for the log hazard ratio).

The estimates with the aforementioned difficulties were removed from the simulation studies. This led to a very small number of estimates being removed and is unlikely to have considerably altered the conclusions. However, it should be noted that the performance of some of the estimators (where estimates were removed) might have been artificially improved by the removal of extreme values. Alternate designs could have been employed to better facilitate small samples and to ensure all generated data were appropriate for the methods being tested. In light of the

difficulties experienced in the simulation studies, one remark is that the bootstrap variance estimator may not always be appropriate for IP weighted analyses in small samples (as one or more of the bootstrap samples may not converge or may give an extreme point estimate leading to an overall high standard error). Likewise, the likelihood-based method for interval-censored data may not always converge or may give large standard errors depending on the combination of left-, interval- and right-censored events in the treatment groups.

The exploratory analysis in Chapter 5 using a sample size of  $n_{obs} = 200$  did still yield interesting results. This work suggested that there might be many scenarios where both the bootstrap and proposed variance estimator are not appropriate for an IP weighted analysis in small samples. A small sample adjustment may need to be considered for use with the proposed M-estimation variance estimator. Further work is warranted to confirm the performance of both the bootstrap and proposed variance estimators in small sample sizes, using appropriate designs for small samples. It would also be useful to investigate what constitutes a “small” and “large” sample size in this context.

A further limitation of the thesis is that often simplified analyses were performed on the datasets. This was because the analyses were for demonstrational purposes rather than to appropriately address a specific research question. Where covariates were available, a more thorough analysis would have included exploring the relationships between the covariates and between the covariates and outcome. Consideration could also have been given as to whether the covariates should be included in the model (for example, if they satisfy the definition of a confounder in the IP weighted analyses), what functional form should be included and whether interactions should be included.

In some cases, it was desired to model the data with a particular distribution. For example, a Weibull model was used in Chapter 3 as the parameters from this analysis were used to inform the Weibull data generating mechanism in the simulation study. In other cases, more thought was given as to the most appropriate choice of distribution. For example, in Chapters 5 and 7 a range of parametric models were

considered and the best model according to the AIC and/or BIC was chosen. More model diagnostics could have been performed to check the fit of the models and to test any assumptions, for example, the proportional hazards assumption. It should be noted that summary statistics (including for the weights in the IP weighted analyses) and an interpretation of the results were still always given in each analysis. It would not have been feasible to perform a rigorous analysis on all five datasets. Further work could include identifying a single dataset that encompasses all three research topics (interval censoring, IP weighting and multistate survival models). This would provide the opportunity to perform a more thorough, unified analysis and to further develop the methods in this thesis so that they can be used simultaneously to address more complex research questions. This is discussed, along with other areas of further work, in the next section.

## 8.6 Further Work

A number of pieces of further work have already been discussed in this chapter. For example, some possible extensions to the simulation studies were given in Section 8.5 and, in particular, the small sample properties of the proposed variance estimator (and bootstrap variance estimator) in an IP weighted analysis should be investigated further. Larger extensions and new pieces of work are proposed in the following paragraphs.

The proposed M-estimation variance estimator for IP weighted parametric models assumed a fixed, binary treatment. One extension to this work could be to allow for more than two treatment groups by including additional estimating equations in the stacked equations and employing multinomial regression to model treatment. Extending the methodology to a continuous or time-varying treatment would require more consideration. Another extension would be to use alternate methods to logistic regression to model the propensity score, such as a probit model, as long as they had a well-defined estimating equation [21]. Similarly, the implementation could be extended to allow for more distributions for the outcome model. In addition,

the proposed estimator could be extended to handle alternate weighting strategies. For example, Hajage *et al* [2] also considered a weight that targeted the average treatment effect in the treated, while Mao *et al* [20] included the matching weight [96] and overlap weight [97].

These extensions could then be incorporated into the accompanying command **stipw**, along with some others. Three main areas of further work for **stipw** could include: checking the balance of confounders between treatment groups, IP weighted Cox models and a wrapper function for postestimation. For example, to check the balance of confounders between treatment groups, summaries of the standardised differences and variance ratio for the raw and weighted data could be given and an overidentification test for covariate balance could be performed. Secondly, **stipw** could be extended to allow for IP weighted Cox models using the theory from Shu *et al* [21]. This would involve maximising the partial likelihood to obtain the variance for the marginal hazard ratio. Obtaining the variance estimate for other marginal estimands, such as survival probabilities and RMST, would be more challenging as it would require estimating the baseline cumulative hazard function and the corresponding uncertainty. Thirdly, a postestimation wrapper function would increase the usability of **stipw** by allowing users to more easily obtain variance estimates of predictions using the M-estimation variance estimator (calculated via the delta method). Finally, the extension of **stipw** to generalised gamma models, frailty and shared-frailty models, relative survival models, cure models and competing risks would constitute larger areas of further work.

This thesis has focused on IP weighting to obtain (contrasts in) marginal predictions. An alternative approach is to use regression standardisation, also known as g-computation. While IP weighting models the treatment, regression standardisation models the outcome. The counterfactuals under each treatment level are then estimated for all individuals and the desired contrast in marginal predictions can be obtained. Chatton *et al* [151] investigated these two approaches in a simulation study. The authors found that, under correct model specification, both methods were unbiased. Variances were obtained via bootstrapping and the authors found

that generally the variance estimation bias ( $100 \times (\text{EmpSE}/\text{ModSE} - 1)$ ) was slightly better for regression standardisation than IP weighting, except when the sample size was small ( $n_{obs} = 100$ ) [151].

M-estimation can be used to estimate the variance in an IP weighted analysis to take into account the associated uncertainty in the estimation of the weights. M-estimation can also be used to estimate the variance in regression standardisation to take into account the associated uncertainty in the covariate distribution [182]. This can be implemented, for example, in **standsurv** [165] in **Stata** for parametric survival models. As far as it is known, the performance of the M-estimation variance estimator for regression standardisation has not been compared with the standard (or bootstrap) variance estimator in a survival setting and this would constitute one area of further work.

Another area of further work could include investigating the performance (both in terms of bias and correct variance estimation) of both methods under departures from the assumptions, as also suggested by Chatton *et al* [151]. For example, IP weighting may be more sensitive to near violations of the positivity assumption, as this may result in more extreme weights. However, the process of performing an IP weighting analysis may be easier than regression standardisation, as there can be fewer considerations with the logistic regression (treatment) model than a survival (outcome) model. For example, for regression standardisation the proportional hazards assumption would need to be verified for all confounders in a proportional hazards model and interactions between the treatment/exposure and the confounders would need to be considered. This may make finding an appropriate model more difficult.

As mentioned in Section 2.6.3, doubly robust estimators are a hybrid of IP weighting and regression standardisation, where both the treatment and outcome are modelled incorporating the confounders [15]. Doubly robust methods can be used to help protect against model misspecification, as they provide a consistent estimate of the target estimand as long as at least one model is correctly specified [15]. However, this consistent estimator may come at the cost of additional sources

of uncertainty. Another piece of further work would be to review doubly robust estimators and their associated variance estimators. There may be scope to then develop an M-estimation variance estimator for doubly robust methods that takes into account both the associated uncertainty in the weight estimation and covariate distribution.

Although the topics of interval censoring, IP weighting and multistate survival models were investigated separately in the thesis, these topics may not appear singly in practice. There are examples in the literature combining two or all three areas. For example, Gran *et al* [183] consider IP weighting (and regression standardisation) in the context of multistate survival models. Considerable research has investigated interval-censored multistate survival models, also known as multistate models for panel data. Examples include the book by van den Hout [184], discussion by Commenges [185] and the `msm` package in R [107]. Gillaizeau *et al* [186] combine all three topics by performing an IP weighted analysis on an illness-death model for interval-censored data. Further work could include an investigation into what methods can be used for this combination of topics, with special consideration to the variance estimator and how predictions are obtained.

A final, and more general, area of further work would be to investigate ways to clearly display the results of simulation studies. Ideally, point estimates and confidence intervals for all scenarios and methods would be displayed on a single graph. However, when there are many scenarios and/or methods, this is not always feasible. The small number of scenarios in Chapter 5 were displayed on two graphs (one each for the stabilised and unstabilised estimators) with confidence intervals using a spike plot. However, it was not feasible to include confidence intervals on the graphs in Chapters 3 and 4. In Chapter 3, nested loop plots were used to display the results from all the scenarios and methods, while line graphs were used in Chapter 4, as many values of one simulation parameter were investigated. Tables for at least some of the scenarios, with MCSE, were given in the relevant appendices for all simulation studies; however, trends are harder to visualise this way. One option is to upload the data into a publicly available repository, so that analysts

can investigate the results themselves, possibly using tools like INTEREST [187] to facilitate the exploration. However, single, graphical summaries are still important and work investigating the best way to display the results from multiple scenarios and methods with uncertainty would be greatly beneficial.

## 8.7 Final Conclusions

In this thesis, parametric survival models have been utilised in three topics in survival analysis: interval censoring, IP weighting and multistate survival models. Methodological research questions have been motivated by example analyses and answered with simulation studies. A novel method to estimate the variance of the parameters in an IP weighted parametric survival model has been proposed and a recently presented method for obtaining predictions from a multistate model [22] has been demonstrated. Alongside the method development, user-friendly software has been provided to facilitate its implementation.

In particular, the key results from the thesis are as follows. Firstly, when performing an analysis on interval-censored data, naive imputation should be avoided in favour of appropriate methods, especially when the interval width is large, differs between groups [114] and/or is irregular [123]. Although this result is unsurprising, end imputation is still frequently used in clinical trials, despite the numerous literature highlighting its limitations. Secondly, when performing an IP weighted analysis on survival data with a fixed, binary treatment, generally either stabilised or unstabilised weights can be used to estimate the marginal treatment effect in the sampled population (if an appropriate variance estimator is used). In terms of variance estimation, either bootstrapping or the proposed M-estimation variance estimator can be used in large samples. The latter can be implemented with the newly written **Stata** command **stipw**. The proposed variance estimator may be advantageous to bootstrapping when computational time or reproducibility are key concerns. Finally, when performing a multistate analysis, the general simulation algorithm proposed by Lambert and Crowther [22] can be used to obtain predictions.

The analysis in the thesis can serve as a reference for this recently proposed method and comparisons of these predictions with non-parametric estimates are facilitated with the majorly redeveloped `msaj` command, which has been extended to provide a comprehensive range of predictions including expected length of stay.

Although not used as frequently as their semi- and non-parametric counterparts, parametric survival models are a powerful tool for analysing survival data. This is especially true with the introduction of Royston-Parmar models, which can capture complex shapes in the hazard function and easily incorporate time-dependent effects. A considerable benefit of parametric models is that they more naturally lend themselves to a range of predictions, such as absolute risks, and these can help provide a more comprehensive summary of the data alongside relative risks. These benefits are why parametric models have been the focus throughout the thesis. Finally, some concluding recommendations are: to consider the use of parametric models and the reporting of absolute risks when performing an analysis on survival data; to clearly present the methods and results of simulation studies using the guidance of Morris *et al* [4] and to provide accompanying software for any newly proposed methods.

# **Appendix A**

---

## **Draft Manuscript for the Closed-form Variance Estimator for IP Weighted Parametric Survival Models**

---

This appendix contains a draft of the paper titled “Variance estimation in inverse probability weighted parametric survival models”, which is to be submitted to a relevant journal.

Statement of contribution: This draft manuscript was in collaboration with my supervisors - Professor Paul Lambert and Dr Michael Crowther. I led the work for this project, developed the novel method, performed the simulation study, implemented the method in the application and performed related coding with input from my supervisors. I drafted the manuscript, which was improved after feedback from both co-authors.

# **Appendix B**

---

## **Manuscript for the Multistate Model Application to the HAI Dataset**

---

The research paper titled “Relaxing the assumption of constant transition rates in a multi-state model in hospital epidemiology” was published in BMC Medical Research Methodology [35] and is available, with supplementary material, via the DOI link: <https://doi.org/10.1186/s12874-020-01192-8>.

Statement of contribution: This manuscript was in collaboration with my supervisors - Professor Paul Lambert and Dr Michael Crowther. I led the work for this project, implemented the methods and coded the analysis with input from my supervisors. I wrote the manuscript, which was improved after feedback from both co-authors.

# Appendix C

---

## Code for `stipw`

---

The user-written command `stipw` in Stata is available on the SSC archive and can be accessed from GitHub at <https://github.com/Micki-Hill/stipw>. This command was completely newly written as part of the work of this thesis. A copy of the code, version 1.0.0, dated 17.01.2022, is given here. `stipw` can be used to obtain M-estimation variance estimates for an IP weighted parametric survival model.

\*! Version 1.0.0 17Jan2022

```
/*
History
MH 17Jan2022: First release to SSC and Github
*/
program define stipw , eclass sortpreserve
    version 15.1
    preserve

    local cmdline : copy local 0

    ** Parse the main command

    syntax [anything] [if] [in] , ///
        Distribution(string) /// Distribution of survival model
    [///
        ANCILLARY /// STREG: Model treatment on the ancillary parameter as well
        OCOEF /// Displays coefficient table from outcome model, before variance is updated (if mest)
        OHEADEr /// Displays header from outcome model, before variance is updated (if mest)
    ///
        BKnots(numlist ascending min=2 max=2) /// STPM2: Boundary knots for baseline
        BKNOTSTVC(numlist ascending min=2 max=2) /// STPM2: Boundary knots for time-dependent
treatment/exposure
        DF(string) /// STPM2: Degrees of freedom for baseline hazard function
        DFTvc(string) /// STPM2: Degrees of freedom for treatment/exposure if time-dependent
        FAILCONVLININIT /// STPM2: Automatically try lininit option if convergence fails
        KNOTS(numlist ascending) /// STPM2: Knot locations for baseline hazard
        KNOTSTVc(numlist ascending) /// STPM2: Knot locations for treatment/exposure if time-dependent
        KNSCALE(string) /// STPM2: Scale for user-defined knots (default scale is time)
        NOORTHog ///
        NOORTHog /// STPM2: Do not use orthogonal transformation of splines variables
        IPWtype(string) /// Weight type - stabilised or unstabilised
        VCE(string) /// Variance type
        GENWeight(string) /// Generates the weights with name string
        GENFlag(string) /// Generates the touse flag with name string
        STSETUpdate ///
        Update the stset call
    ///
        NOHEADer /// Suppress header from final coefficient table
        NOHR /// STREG: Do not report hazard ratios
        TRatio ///
        NOShow /// STREG: Do not show st setting information from outcome model, before variance is updated
(if mest)
        EForm ///
        ALLEQ ///
        KEEPCons ///
        SHOWCons ///
    ///
        NOLOg ///
        LINinit ///
        * ///
        ML and display options
    ]
marksample touse
mlopts out_mlopts options, `options'
_get_diopts diopts, `options'
local onolog : copy local nolog

* Parse each model, separated by brackets, should just be one treatment model
_parse expand trt_model op : anything
```

```

* Extract out the logit part
gettken tcmd 0 : trt_model_1

* Syntax for the treatment model
syntax varlist(min=2 numeric) [, ///
    NOCONSTant ///
    OFFset(varname numeric) ///
    TCOEF ///
    NOLog ///
    * ///
]
mlopts trt_mlopts, `options'

* Parse the varlist to get treatment and confounders
gettken trt Z : varlist

** Error checks: Treatment model

* Check one treatment model is given and in brackets
gettken _ignore : anything, match(brackets)
if ("`brackets'" != "(") {
    di as error "Treatment model needs to be enclosed in brackets"
    exit 198
}
if ("`trt_model_2'" != "") {
    di as error "Only one treatment model should be given. It should be enclosed in brackets."
    exit 198
}

* Check the first word is logit
if ("`tcmd'" != "logit") {
    di as error "Logit needs to be specified first in the treatment model"
    exit 198
}

* Treatment variable should only have variables 0 or 1 (or missing)
cap assert inlist('trt',0,1,.)
if (_rc != 0) {
    di as error "Treatment variable should be a binary variable with values 0 and 1."
    exit 198
}

* Treatment should have both 0 and 1
cap assert inlist('trt',0,.)
if (_rc == 0) {
    di as error "Treatment variable should have both values 0 and 1, currently all values are 0 (or missing)."
    exit 198
}
cap
cap assert inlist('trt',1,.)
if (_rc == 0) {
    di as error "Treatment variable should have both values 0 and 1, currently all values are 1 (or missing)."
    exit 198
}
cap

* Logit only supported
if ("`tdistribution'" != "logit" & "`tdistribution'" != "") {
    di as error "Only logit is currently supported for treatment model."
}

```

```

    exit 198
}

** Error checks: Outcome model

* Check data has been stset
cap st_is 2 analysis
if (_rc != 0) {
    di as error "Data must be stset"
    exit 198
}

* Check weights have not already been specified
if "`_dta[st_w]'" != "" {
    di as error "Data should be stset without weights."
    exit 198
}

* Check st has not been set with id variable (multiple weights not allowed)
if "`_dta[st_id]'" != "" {
    di as error "Data should be stset without id option: multiple-record-per-subject survival data are not
supported."
    exit 198
}

** Error checks: Outcome model - streg

* Check distribution is one that is currently supported
local l = length("`distribution`")
if substr("exponential",1,'l') != "`distribution`" & ///
    substr("weibull",1,'l') != "`distribution`" & ///
    substr("gompertz",1,max(3,'l')) != "`distribution`" & ///
    substr("lognormal",1,max(4,'l')) != "`distribution`" & ///
    substr("Inormal",1,max(2,'l')) != "`distribution`" & ///
    substr("loglogistic",1,max(4,'l')) != "`distribution`" & ///
    substr("llogistic",1,max(2,'l')) != "`distribution`" & ///
    "`distribution`" != "rp" {
    di as error "Currently exponential, Weibull, Gompertz, log-logistic, log-normal and rp survival models are
supported."
    exit 198
}

* Check ancillary not specified with exponential model
if (substr("exponential",1,'l') == "`distribution`" & "`ancillary`" != "") {
    di as error "Ancillary option not allowed with the exponential model"
    exit 198
}

* Check nohr only specified with Exp, Weib and Gompertz
if ("`nohr'" != "" & substr("exponential",1,'l') != "`distribution`" & ///
    substr("weibull",1,'l') != "`distribution`" & ///
    substr("gompertz",1,max(3,'l')) != "`distribution`") {
    di as error "Option nohr only allowed with exponential, Weibull and Gompertz distributions"
    exit 198
}

* Check tratio only specified with Lognormal, Loglogistic, Weibull, exponential
if ("`tratio'" != "" & substr("gompertz",1,max(3,'l')) == "`distribution`") {
    di as error "Option tratio only allowed with exponential, Weibull, log-normal and log-logistic"
}

```

```

    exit 198
}

* Check only one of nohr and tratio is specified
if ("`nohr'" != "" & "`tratio'" != "") {
    di as error "Only one of nohr and tratio is allowed"
    exit 198
}

* Check stpm2 options are not specified with streg
if ("`distribution'" != "rp" & ("`bknknotstvc'" != "" | "`bknotstvc'" != "" | ///
    "`df'" != "" | "`dftvc'" != "" | "`failconvlinit'" != "" | "`knots'" != "" | ///
    "`knotstvc'" != "" | "`knscale'" != "" | "`noorthog'" != "" | "`eform'" != "" | ///
    "`alleq'" != "" | "`keepcons'" != "" | "`showcons'" != "" | "`lininit'" != ""))
{
    di as error "The following options are not permitted with streg models:"
    di as error "bknknotstvc, bknotstvc, df, dftvc, failconvlinit, knots, knotstvc knscale, noorthorg, eform, alleq, keepcons, showcons, lininit"
    exit 198
}

** Error checks: Outcome model - stpm2

* Dftvc must be a number
if (`dftvc' != "") {
    cap confirm integer number `dftvc'
    if _rc>0 {
        display as error "dftvc option must be an integer"
        exit 198
    }
}

* Bknotstvc can only be specified if dftvc or knotstvc is specified
if ("`dftvc'" == "" & "`knotstvc'" == "" & "`bknotstvc'" != "") {
    di as error "bknotstvc can only be specified if dftvc or knotstvc is also specified"
    exit 198
}

* Check streg options not specified with stpm2
if ("`distribution'" == "rp" & ("`ancillary'" != "" | ///
    "`nohr'" != "" | "`tratio'" != "" | "`noshow'" != ""))
{
    di as error "The following options are not permitted with rp (stpm2) models:"
    di as error "ancillary, nohr, tratio, noshow"
    exit 198
}

** Other error checks

* Check weight is only one of unstabilised or stabilised and set to stabilised/unstabilised
if ("`ipwtype'" != "") {
    local lw = length("`ipwtype'")
    if substr("unstabilised",1,'lw') != "`ipwtype'" & substr("stabilised",1,'lw') != "`ipwtype'" {
        di as error "Only stabilised and unstabilised weights are allowed"
        exit 198
    }
    else {
        if substr("unstabilised",1,'lw') == "`ipwtype'" {
            local ipwtype = "unstabilised"
        }
        else {

```

```

        local ipwtype = "stabilised"
    }
}
else {
    local ipwtype = "stabilised"
}

* Check weight variable not already defined
cap confirm variable `genweight'
if (_rc == 0 & "`genweight'" != "") {
    di as error "Variable `genweight' already exists."
    exit 198
}
cap

* Check flag variable not already defined
cap confirm variable `genflag'
if (_rc == 0 & "`genflag'" != "") {
    di as error "Variable `genflag' already exists."
    exit 198
}
cap

* Check variance one of Mestimation or robust. Assign to mestimation if missing
if (!inlist("`vce'", "", "robust", "mestimation")) {
    di as error "Variance type must be robust or mestimation"
    exit 198
}
if ("`vce'" == "") {
    local vce "mestimation"
}

** Missing data

* Flag variable
if ("`genflag'" == "") {
    cap drop _stipw_flag
    cap
    local genflag = "_stipw_flag"
}

tempname miss
qui egen `miss' = rowmiss(`trt' `Z') if `touse'
qui replace `miss' = `miss' + 1 if `touse' & _st == 0
qui count if `miss' > 0 & `touse'
if r(N) > 0 {
    di "`r(N)' observations have missing treatment and/or missing confounder values and/or _st = 0."
    di "These observations are excluded from the analysis, see variable `genflag'"
    di ""
    qui replace `touse' = 0 if `miss' > 0 & `touse'
}
qui gen `genflag' = `touse'

** Fit the models

* Treatment model
di "Fitting logistic regression to obtain denominator for weights"
if ("`tcoef'" == "") {

```

```

    local tnocoef = "nocoef"
}
if ("`offset'" != "") {
    logit `trt' `Z' if `genflag', `noconstant' offset(`offset') `tnocoef' `nolog' `trt_mlopts'
}
else {
    logit `trt' `Z' if `genflag', `noconstant' `tnocoef' `nolog' `trt_mlopts'
}
tempname alphas
mat `alphas' = e(b)
local cmdline_logit = e(cmdline)
local rc_logit = e(rc)
local converged_logit = e(converged)

* Weights for denominator
tempvar ps
qui predict `ps' if `genflag'

* Stabilised weights only - second logit for numerator of weights
if ("`ipwtype'" == "stabilised") {
    di "Fitting second logistic regression with no confounders to obtain numerator for stabilised weights"
    logit `trt' if `genflag', `tnocoef' `nolog'
    tempname alphas2
    mat `alphas2' = e(b)
    matrix coleq `alphas2' = ``trt'2"
    local cmdline_logit2 = e(cmdline)
    local rc_logit2 = e(rc)
    local converged_logit2 = e(converged)
}

* Weights
if ("`genweight'" == "") {
    cap drop _stipw_weight
    cap
    local genweight = "_stipw_weight"
}
if ("`ipwtype'" == "stabilised") {
    tempvar prev
    qui predict `prev' if `genflag'
    qui gen double `genweight' = `trt'*`prev'/`ps' + (1-`trt')*(1-`prev')/(1-`ps') if `genflag'
}
else {
    qui gen double `genweight' = `trt'/'ps' + (1-`trt')/(1-`ps') if `genflag'
}

* Outcome model
di ""
di "Fitting weighted survival model to obtain point estimates"

qui streset [pw = `genweight'] if `genflag'

if ("`ancillary'" != "") {
    local anc = "ancillary(`trt')"
}
if ("`ocoef'" == "") {
    local onocoef = "nocoef"
}

```

```

if ("`oheader'" == "") {
    local onoheader = "noheader"
}

if ("`distribution'" == "rp") {
    foreach option in bknots knots knscale dftvc {
        if ("`option'" != "") {
            local `option'_ = "`option'(`option')"
        }
    }

    if ("`dftvc'" != "" | "`knotstvc'" != "") {
        local tvc = "tvc(`trt')"
        foreach option in bknotstvc knotstvc {
            if ("`option'" != "") {
                local `option'_ = "`option'(`trt' `option')"
            }
        }
    }
}

local cmd = "stpm2"
stpm2 `trt' if `genflag', df(`df') scale(hazard) `bknots_` `knots_` `knscaler_` `tvc` `dftvc_` `bknotstvc_` `knotstvc_` `onolog` `lininit` `failconvlinit` `noorthog` `onoheader` `onocoef` `keepcons` `out_mlopts`

tempvar xb dxb
qui predict `xb' if `genflag', xb
qui predict `dxb' if `genflag', dx

local rcs = e(rcsterms_base)
local drcs = e(drcsterms_base)
if ("`tvc'" != "") {
    local rcs_trt = e(rcsterms_`trt')
    local drcs_trt = e(drcsterms_`trt')
}

local del_entry = e(del_entry)
if (`del_entry' == 1) {
    tempvar xb0
    qui predict `xb0' if `genflag', xb timevar(_t0)

    local dfbase = e(dfbase)
    local s0_rcs_s0_rcs1
    forvalues i = 2/`dfbase' {
        local s0_rcs `s0_rcs'_`s0_rcs`i'
    }
    if ("`tvc'" != "") {
        local df_trt = e(df_`trt')
        local s0_rcs_trt_s0_rcs_trt1
        forvalues i=2/`df_`trt'' {
            local s0_rcs_trt `s0_rcs_trt'_`s0_rcs_`trt`i'
        }
    }
}

else {
    local cmd = "streg"
    streg `trt' if `genflag', d(`distribution') `anc' `onoheader` `onocoef` `onolog` `noshow` `out_mlopts'
}

local model = "`e(cmd)'"

```

```

tempname betas betas_unique Vmodel
mat `betas' = e(b)
local rank = e(rank)
matselrc `betas' `betas_unique' , c(1/`rank')
if ("`ipwtype'" == "stabilised") {
    mat b_full = `alphas', `alphas2', `betas_unique'
}
else {
    mat b_full = `alphas', `betas_unique'
}
mat robust = e(V)
mat `Vmodel' = e(V_modelbased)

if ("`stsetupdate'" == "") {
    tempfile new_variables
    tempvar id
    qui gen `id' = _n
    qui save `new_variables'
}

** Get M-estimates for variance if needed

if ("`vce'" == "mestimation") {
    mata: stipw()
}

** Store results

* Return codes and convergence
ereturn scalar rc_logit = `rc_logit'
ereturn scalar converged_logit = `converged_logit'
if ("`ipwtype'" == "stabilised") {
    ereturn scalar rc_logit2 = `rc_logit2'
    ereturn scalar converged_logit2 = `converged_logit2'
}

* Treatment var and counts on and off treatment
ereturn local tvar = "`trt'"
qui count if `trt' == 0 & `genflag'
ereturn scalar n0 = r(N)
qui count if `trt' == 1 & `genflag'
ereturn scalar n1 = r(N)

* Command and command lines
ereturn local cmdline_`cmd' = strtrim(strtrim(e(cmdline)))
ereturn local cmdline = "stipw `cmdline'"
ereturn local cmdline_logit = strtrim(strtrim("`cmdline_logit'"))
if ("`ipwtype'" == "stabilised") {
    ereturn local cmdline_logit2 = strtrim(strtrim("`cmdline_logit2'"))
}
ereturn local cmd3 = "stipw"

* Weight type
ereturn local ipwtype = "`ipwtype'"

* Offset for logit
if ("`offset'" != "") {
    ereturn local offset_logit = "`offset'"
}

```

```

** Store results: M-estimation only
if "`vce'" == "mestimation" {

    ereturn local vcetype = "M-estimation"
    ereturn local vce = "mestimation"

    local names = `:colname b_full'
    local eqs = `:coleq b_full'
    foreach matrix in var_full A B {
        matrix rownames `matrix' = `names'
        matrix colnames `matrix' = `names'
        matrix roweq `matrix' = `eqs'
        matrix coleq `matrix' = `eqs'
    }

    ereturn repost V = var_out , esample(`touse')
    ereturn matrix V_B = B
    ereturn matrix V_A = A
    ereturn matrix V_robust = robust
    ereturn matrix V_full = var_full
    ereturn matrix b_full = b_full

    if "`ancillary'" != "" {
        ereturn scalar df_m = 2
    }

    * Redo Wald test
    qui testparm *`trt'*'
    ereturn scalar chi2 = r(chi2)
    ereturn scalar p = r(p)
    ereturn local chi2type = "Wald"
}

```

## \*\* Display results

```

di ""
di "Displaying weighted survival model with `e(vcetype)' standard errors"

if ("`nohr'" == "" & "`distribution'" != "rp") {
    local trans = "hr"
}
if ("`tratio'" != "") {
    local trans = "tratio"
}
if ("`alleq'" == "" & "`distribution'" == "rp") {
    local neq = "neq(1)"
}
if ("`showcons'" == "" & "`distribution'" == "rp") {
    local nocnsreport = "nocnsreport"
}
ml display , `trans' `eform' `noheader' `neq' `nocnsreport' `diopts'

```

## \*\* Save the weights and touse

```

if ("`stsetupdate'" != "") {
    restore, not
    di ""
}

```

```

        di "Warning: stset has been updated with the weights"
    }

else {
    * Clear the data characteristics from the using data
    use `new_variables', replace
    local ilist: char _dta[]
    foreach i in `ilist' {
        char _dta[`i']
    }
    qui save `new_variables', replace

    * Clear any stipw/stpm2 variables that will be replaced
    restore
    if ("`genweight'" == "_stipw_weight") {
        cap drop _stipw_weight
        cap
    }
    if ("`genflag'" == "_stipw_flag") {
        cap drop _stipw_flag
        cap
    }
    if ("`distribution'" == "rp") {
        cap drop _rcs*
        cap
        cap drop _d_rcs*
        cap
        cap drop _s0_rcs*
        cap
    }
    * Merge new variables into the dataset
    qui gen `id' = _n
    qui merge 1:1 `id' using `new_variables', noreport nogen ///
        keepusing(`id' `genweight' `genflag' `rcs' `drcs' `rcs_trt' `drcs_trt' `s0_rcs' `s0_rcs_trt')

    * Update e(sample)
    tempvar samp
    qui gen `samp' = `genflag'
    ereturn repost b = `betas', esample(`samp')
}

end

```

```

version 15.1
set matastrict on
mata:

```

```

///////////
// Define structure //
///////////

struct stipw_info {

// Importing data
    real matrix    touse,          // Marker for [in] and [if]
                t,              // Survival time
                t0,             // Entry time
                d,              // Event indicator

```

```

trt,           // Treatment indicator
Z_,            // Covariates without intercept
Z,             // Covariates with intercept
offsetvar,     // Offset variable if specified
rcs,           // Restricted cubic spline variables
drcs,          // Differential of restricted cubic spline variables
rcs_trt,       // Restricted cubic spline variables for trt
drcs_trt,      // Differential of restricted cubic spline variables for trt
s0_rcs,        // Restricted cubic spline variables for delayed entry
s0_rcs_trt,    // Restricted cubic spline variables for trt for delayed entry
xb,            // Predicted xb from stpm2
xb0,           // Predicted xb from stpm2 with timevar t0
dxb            // Predicted dxb from stpm2

// Parameter estimates
real matrix alphas,           // Point estimates for the treatment model
alphas2,        // Point estimate for the second treatment model, intercept for logit model for trt
with no covariates
with no covariates
betas,          // Point estimates for the outcome model
thetas,         // All parameters
Vmodel,         // Model based variance matrix, used as part of A matrix (issue for Weibull, so done
manually)

// Counts
real scalar stab,             // Indicator for stabilised weights 1 = stabilised, 0 = unstabilised
nocons,          // Exclude constant term in treatment model 1 = Yes
offset,          // Offset variable for treatment model 1 = Yes
anc,             // Treatment modelled on ancillary parameter (streg) 1 = Yes
tvc,             // Treatment modelled as tvc (stpm2) 1 = Yes
del,             // Delayed entry (stpm2) 1 = Yes
n,               // Number of patients
nalphas,         // Number of treatment parameters
nbetas,          // Number of outcome parameters
nthetas,         // Total number of parameters

// Outcome model options
string scalar model           // Type of outcome model

// Pointers
pointer(real scalar function) matrix score,      // score function (u function in M-estimation)
hessian          // Hessian matrix - derivative of score/u function

}

///////////////////////////////
// Fill in structure //
////////////////////////////

function stipw_get_stuff() {
    struct stipw_info scalar S
    stipw_get_stuff_general(S)
    return(S)
}

function stipw_get_stuff_general(struct stipw_info scalar S) {

// Get details about command
S.model           = st_local("model")
}

```

```

// Read in data
S.touse          = st_local("touse")
S.stab           = st_local("ipwtype") == "stabilised"
S.nocons         = st_local("noconstant") != ""
S.offset         = st_local("offset") != ""
S.anc            = st_local("ancillary") != ""
S.tvc            = st_local("tvc") != ""
S.del            = st_local("del_entry") == "1"

S.t              = st_data(., "_t", S.touse)
S.t0             = st_data(., "_t0", S.touse)
S.d              = st_data(., "_d", S.touse)
S.trt            = st_data(., st_local("trt"), S.touse)
S.Z_             = st_data(., st_local("Z"), S.touse)
if (S.offset) S.offsetvar = st_data(., st_local("offset"), S.touse)

if (S.model == "stpm2") {
    S.rcs          = st_data(., st_local("rcs"), S.touse)
    S.drcs         = st_data(., st_local("drcs"), S.touse)
    S.xb           = st_data(., st_local("xb"), S.touse)
    S.dxb          = st_data(., st_local("dxb"), S.touse)
    if (S.tvc == 1) {
        S.rcs_trt   = st_data(., st_local("rcs_trt"), S.touse)
        S.drcs_trt   = st_data(., st_local("drcs_trt"), S.touse)
    }
    if (S.del == 1) {
        S.s0_rcs     = st_data(., st_local("s0_rcs"), S.touse)
        if (S.tvc == 1) {
            S.s0_rcs_trt = st_data(., st_local("s0_rcs_trt"), S.touse)
        }
        S.xb0          = st_data(., st_local("xb0"), S.touse)
    }
}

S.alphas         = st_matrix(st_local("alphas"))
if (S.stab) S.alphas2 = st_matrix(st_local("alphas2"))
S.betas          = st_matrix(st_local("betas"))
S.Vmodel         = st_matrix(st_local("Vmodel"))

if (S.model == "stpm2") {
    if (S.del == 1) S.betas = S.betas[1::(cols(S.betas)+2)/3]
    else S.betas = S.betas[1::cols(S.betas)/2+1]
    S.Vmodel = S.Vmodel[1::cols(S.betas), 1::cols(S.betas)]
}
if (S.stab) S.thetas = S.alphas, S.alphas2, S.betas
else S.thetas = S.alphas, S.betas

S.n              = rows(S.t)
S.nthetas        = cols(S.thetas)
S.nalphas        = cols(S.alphas)
S.nbetas         = cols(S.betas)

if (S.nocons) S.Z_ = S.Z_
else S.Z_ = S.Z_, J(S.n, 1, 1)
}

///////////////////////////////
// Declare pointers //
/////////////////////////////

```

```

void function stipw_declare_pointers(struct stipw_info scalar S)
{
    // Exponential
    if (S.model == "ereg") {
        S.score = &stipw_exp_score()
    }

    // Weibull
    if (S.model == "weibull") {
        if (S.anc) {
            S.score = &stipw_weibull_anc_score()
            S.hessian = &stipw_weibull_anc_hessian()
        }
        else {
            S.score = &stipw_weibull_score()
            S.hessian = &stipw_weibull_hessian()
        }
    }

    // Gompertz
    if (S.model == "gompertz") {
        if (S.anc) S.score = &stipw_gompertz_anc_score()
        else S.score = &stipw_gompertz_score()
    }

    // Log-logistic
    if (S.model == "llogistic") {
        if (S.anc) S.score = &stipw_loglogistic_anc_score()
        else S.score = &stipw_loglogistic_score()
    }

    // Log normal
    if (S.model == "Inormal") {
        if (S.anc) S.score = &stipw_lognormal_anc_score()
        else S.score = &stipw_lognormal_score()
    }

    // stpm2
    if (S.model == "stpm2") S.score = &stipw_stpm2_hazard_score()
}

///////////////////////////////
//      stipw      //
/////////////////////////////

```

```

void function stipw()
{
    real matrix lp,          // Confounders (z) multiplied by alphas (a) with offset included
         ps,          // Estimated propensity score
         prev,         // Prevalence of treatment (stabilised weights)
         w,           // Estimated (unstabilised or stabilised) weight from PS
         diffw,        // Differential of the weights wrt to the alphas
         diffw2,       // Differential of the weights wrt to the alphas2
         uout,         // u (or score) function for the outcome model without weights
         u,            // u function for all paramters (trt and out)
         H,            // Derivative of u (or Hessian) for the outcome model without weights
         A,            // A matrix of the sandwich estimator
         invA,         // Inverse of A
         B,            // B matrix of the sandwich estimator
}

```

```

var_full,          // Variance from M-estimation
var_out,          // Variance matrix of the survival model only
var_stpm          // Variance matrix with additional equations for stpm2

real scalar       nstpm,           // Number of parameters in full stpm2 variance (no delayed entry)
                  nstpmd,          // Number of parameters in full stpm2 variance (delayed entry)

// Define structure
struct stipw_info scalar S

// Setting up
S = stipw_get_stuff()
stipw_declare_pointers(S)

// Create the weights
if (S.offset) lp      = (S.Z,S.offsetvar)*(S.alphas,1)'
else lp               = S.Z*S.alphas'
ps                   = (1:+exp(-1:*lp)):^( -1)
if (S.stab) {
    prev              = (1+exp(-S.alphas2))^( -1)
    w                 = S.trt :* prev :/ ps :+ (1 :- S.trt) :* (1 :- prev) :/ (1 :- ps)
}
else w               = S.trt :/ ps :+ (1 :- S.trt) :/ (1 :- ps)

// Get the u (for outcome) and Hessian
uout                = (*S.score)(S)
if (S.model == "weibull") {
    H                 // Issue with unweighted variance for Weibull, so done manually
    = rowshape((-1/S.n * quadcolsum(w :* (*S.hessian)(S))),S.nbetas)
}
else H               = qrinv(S.n :* S.Vmodel)

// Matrix A
A                   = J(S.nthetas,S.nthetas,0)

// Matrix A for trt model
A[(1::S.nalphanas),(1::S.nalphanas)] = 1/S.n :* S.Z' * (S.Z :* exp(lp) :/ (exp(lp) :+ 1):^2)

// Matrix A for 2nd trt model (stabilised weights)
if (S.stab) {
    A[S.nalphanas+1,S.nalphanas+1] = exp(S.alphas2)/((exp(S.alphas2)+1)^2)
}

// Matrix A for out model
A[(S.nthetas-S.nbetas+1::S.nthetas),(S.nthetas-S.nbetas+1::S.nthetas)] = H

// Matrix A for trt model and 2nd trt model (stabilised weights): 0 matrix, independent

// Matrix A for trt and out model, lower left rectangle
if (S.stab) diffw      = S.Z :* (exp(lp) :* (1 - prev) :- S.trt :* (exp(-1 :* lp) :* prev + exp(lp) :* (1 - prev)))
else        diffw      = S.Z :* (exp(lp) :- S.trt :* (exp(-1 :* lp) + exp(lp) ))
A[(S.nthetas-S.nbetas+1::S.nthetas),(1::S.nalphanas)] = -1/S.n * uout'*diffw

// Matrix A for 2nd trt and out model (stabilised weights)
if (S.stab) {
    diffw2 = S.trt :* exp(-S.alphas2) :/ ((1+exp(-S.alphas2))^2) :/ ps :- (1 :- S.trt) :* exp(S.alphas2) :/
((1+exp(S.alphas2))^2) :/ (1 :- ps)
A[(S.nthetas-S.nbetas+1::S.nthetas),S.nalphanas+1] = -1/S.n * uout'*diffw2
}

```

```

}

st_matrix("A",A)

// Matrix B
if (S.stab) u = S.Z:(S.trt:-ps), S.trt :- prev, w:*uout
else u = S.Z:(S.trt:-ps), w:*uout
B = 1/S.n :* u'*u
st_matrix("B",B)

// Get variance
invA = qrinv(A)
var_full = 1/S.n :* invA*B*(invA')
st_matrix("var_full",var_full)

var_out = var_full[(S.nthetas-S.nbetas+1::S.nthetas),(S.nthetas-S.nbetas+1::S.nthetas)]
if (S.model == "stpm2") {

    nstpm = S.nbetas*2-2
    var_stpm = J(nstpm, nstpm,.)
    if (S.del == 1) {
        nstpmd = S.nbetas*3-2
        var_stpm = J(nstpmd, nstpmd,.)
    }

    var_stpm[(1::S.nbetas),(1::S.nbetas)] = var_out
    var_stpm[(1::S.nbetas),(S.nbetas+1::nstpm)] = var_out[,,(2::S.nbetas-1)]
    var_stpm[(S.nbetas+1::nstpm),(1::S.nbetas)] = var_out[(2::S.nbetas-1),]
    var_stpm[(S.nbetas+1::nstpm),(S.nbetas+1::nstpm)] = var_out[(2::S.nbetas-1),(2::S.nbetas-1)]

    if (S.del == 1) {
        var_stpm[(2*S.nbetas-1)::nstpmd,(1::S.nbetas)] = var_out
        var_stpm[(1::S.nbetas),(2*S.nbetas-1)::nstpmd] = var_out
        var_stpm[(2*S.nbetas-1)::nstpmd,(2*S.nbetas-1)::nstpmd] = var_out
        var_stpm[(2*S.nbetas-1)::nstpmd,(S.nbetas+1::nstpm)] = var_out[,,(2::S.nbetas-1)]
        var_stpm[(S.nbetas+1::nstpm),(2*S.nbetas-1)::nstpmd] = var_out[(2::S.nbetas-1),]
    }

    var_out = var_stpm
}
st_matrix("var_out",var_out)

}

///////////////////////////////
// streg - Exponential //
////////////////////////////

// Score function
function stipw_exp_score(struct stipw_info scalar S)
{
    real scalar loghr, loglambda
    real matrix tlp, q, tlp0, A1, A2

    loghr = S.betas[1]
    loglambda = S.betas[2]

    tlp = S.t :* exp(loglambda :+ loghr :* S.trt)           // t*exp(linear predictor)
}

```

```

A1      = S.trt :* (S.d :- tlp)
A2      = S.d :- tlp

q      = selectindex(S.t0)
if (rows(q) > 0) {
    tlp0  = S.t0 :* exp(loglambda :+ loghr :* S.trt)           // delayed entry

    A1[q] = A1[q] :+ S.trt[q] :* tlp0[q]
    A2[q] = A2[q] :+ tlp0[q]
}

return(A1, A2)
}

```

```

///////////////
// streg - Weibull //
/////////////

```

```

// Score function: No ancillary parameter
function stipw_weibull_score(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, gamma
    real matrix tglp, lt, q, tglp0, lt0, A1, A2, A3

    loghr      = S.betas[1]
    loglambda   = S.betas[2]
    gamma       = exp(S.betas[3])

    tglp        = S.t :^ gamma :* exp(loglambda :+ loghr :* S.trt)
    lt          = log(S.t)

    A1          = S.trt :* (S.d :- tglp)
    A2          = S.d :- tglp
    A3          = S.d :* (1 :+ gamma :* lt) :- gamma :* lt :* tglp

    q      = selectindex(S.t0)
    if (rows(q) > 0) {
        tglp0  = S.t0 :^ gamma :* exp(loglambda :+ loghr :* S.trt)
        lt0    = log(S.t0)

        A1[q] = A1[q] :+ S.trt[q] :* tglp0[q]
        A2[q] = A2[q] :+ tglp0[q]
        A3[q] = A3[q] :+ gamma :* lt0[q] :* tglp0[q]
    }

    return(A1, A2, A3)
}

```

```

// Score function: Ancillary parameter
function stipw_weibull_anc_score(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, loganc, loggamma
    real matrix anc, tglp, lt, q, tglp0, lt0, A1, A2, A3, A4

    loghr      = S.betas[1]
    loglambda   = S.betas[2]
    loganc     = S.betas[3]
    loggamma   = S.betas[4]

```

```

anc      = exp(loggamma :+ loganc :* S.trt)
tglp    = S.t :^ anc :* exp(loglambda :+ loghr :* S.trt)
lt      = log(S.t)

A1      = S.trt :* (S.d :- tglp)
A2      = S.d :- tglp
A3      = S.trt :* (S.d :* (1 :+ anc :* lt) :- anc :* lt :* tglp)
A4      = S.d :* (1 :+ anc :* lt) :- anc :* lt :* tglp

q      = selectindex(S.t0)
if (rows(q) > 0) {
    tglp0 = S.t0 :^ anc :* exp(loglambda :+ loghr :* S.trt)
    lt0   = log(S.t0)

    A1[q] = A1[q] :+ S.trt[q] :* tglp0[q]
    A2[q] = A2[q] :+ tglp0[q]
    A3[q] = A3[q] :+ S.trt[q] :* anc[q] :* lt0[q] :* tglp0[q]
    A4[q] = A4[q] :+ anc[q] :* lt0[q] :* tglp0[q]
}

return(A1, A2, A3, A4)
}

```

```

// Hessian function: No ancillary parameter
function stipw_weibull_hessian(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, gamma
    real matrix tglp, lt, glt, gdiff, q, tglp0, lt0, glt0, gdiff0
    real matrix A11, A12, A13, A22, A23, A33

    loghr      = S.betas[1]
    loglambda   = S.betas[2]
    gamma       = exp(S.betas[3])

    tglp        = S.t :^ gamma :* exp(loglambda :+ loghr :* S.trt)
    lt          = log(S.t)
    glt         = gamma :* lt
    gdiff       = glt :* tglp

    A11        = -1 :* S.trt:^2 :* tglp
    A12        = -1 :* S.trt :* tglp
    A13        = -1 :* S.trt :* gdiff
    A22        = -1 :* tglp
    A23        = -1 :* gdiff
    A33        = S.d :* glt :- gdiff :* (glt :+ 1)

    q      = selectindex(S.t0)
    if (rows(q) > 0) {
        tglp0 = S.t0 :^ gamma :* exp(loglambda :+ loghr :* S.trt)
        lt0   = log(S.t0)
        glt0   = gamma :* lt0
        gdiff0 = glt0 :* tglp0

        A11[q] = A11[q] :+ S.trt[q]:^2 :* tglp0[q]
        A12[q] = A12[q] :+ S.trt[q] :* tglp0[q]
        A13[q] = A13[q] :+ S.trt[q] :* gdiff0[q]
        A22[q] = A22[q] :+ tglp0[q]
        A23[q] = A23[q] :+ gdiff0[q]
        A33[q] = A33[q] :+ gdiff0[q] :* (glt0[q] :+ 1)
    }
}

```

```

}

return(A11, A12, A13, A12, A22, A23, A13, A23, A33)
}

// Hessian function: Ancillary parameter
function stipw_weibull_anc_hessian(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, loganc, loggamma
    real matrix anc, tglp, lt, glt, gdiff, q, tglp0, lt0, glt0, gdiff0
    real matrix A11, A12, A13, A14, A22, A23, A24, A33, A34, A44

    loghr      = S.betas[1]
    loglambda   = S.betas[2]
    loganc     = S.betas[3]
    loggamma    = S.betas[4]

    anc        = exp(loggamma :+ loganc :* S.trt)
    tglp       = S.t :^ anc :* exp(loglambda :+ loghr :* S.trt)
    lt         = log(S.t)
    glt        = anc :* lt
    gdiff      = glt :* tglp

    A11        = -1 :* S.trt:^2 :* tglp
    A12        = -1 :* S.trt :* tglp
    A13        = -1 :* S.trt:^2 :* gdiff
    A14        = -1 :* S.trt :* gdiff
    A22        = -1 :* tglp
    A23        = -1 :* S.trt :* gdiff
    A24        = -1 :* gdiff
    A33        = S.trt :^ 2 :* (S.d :* glt :- gdiff :* (glt :+ 1))
    A34        = S.trt :* (S.d :* glt :- gdiff :* (glt :+ 1))
    A44        = S.d :* glt :- gdiff :* (glt :+ 1)

    q          = selectindex(S.t0)
    if (rows(q) > 0) {
        tglp0  = S.t0 :^ anc :* exp(loglambda :+ loghr :* S.trt)
        lt0    = log(S.t0)
        glt0   = anc :* lt0
        gdiff0 = glt0 :* tglp0

        A11[q]  = A11[q] :+ S.trt[q]:^2 :* tglp0[q]
        A12[q]  = A12[q] :+ S.trt[q] :* tglp0[q]
        A13[q]  = A13[q] :+ S.trt[q]:^2 :* gdiff0[q]
        A14[q]  = A14[q] :+ S.trt[q] :* gdiff0[q]
        A22[q]  = A22[q] :+ tglp0[q]
        A23[q]  = A23[q] :+ S.trt[q] :* gdiff0[q]
        A24[q]  = A24[q] :+ gdiff0[q]
        A33[q]  = A33[q] :+ S.trt[q]:^2 :* gdiff0[q] :* (glt0[q] :+ 1)
        A34[q]  = A34[q] :+ S.trt[q] :* gdiff0[q] :* (glt0[q] :+ 1)
        A44[q]  = A44[q] :+ gdiff0[q] :* (glt0[q] :+ 1)
    }

    return(A11, A12, A13, A14, A12, A22, A23, A24, A13, A23, A33, A34, A14, A24, A34, A44)
}
}

///////////////////////////////
// streg - Gompertz //
/////////////////////////////

```

```

// Score function: No ancillary parameter
function stipw_gompertz_score(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, gamma
    real matrix lp, gt, gS, gSdiff, q, gt0, gSO, gSdiff0, A1, A2, A3

    loghr      = S.betas[1]
    loglambda  = S.betas[2]
    gamma      = S.betas[3]

    lp          = exp(loglambda :+ loghr :* S.trt)
    gt          = gamma :* S.t
    gS          = lp :* gamma^(-1) :* (exp(gt) :- 1)
    gSdiff     = lp :* gamma^(-2) :* (exp(gt) :* (gt :- 1) :+ 1)

    A1          = S.trt :* (S.d :- gS)
    A2          = S.d :- gS
    A3          = S.d:*S.t :- gSdiff

    q           = selectindex(S.t0)
    if (rows(q) > 0) {
        gt0        = gamma :* S.t0
        gSO        = lp :* gamma^(-1) :* (exp(gt0) :- 1)
        gSdiff0   = lp :* gamma^(-2) :* (exp(gt0) :* (gt0 :- 1) :+ 1)

        A1[q]      = A1[q] :+ S.trt[q] :* gSO[q]
        A2[q]      = A2[q] :+ gSO[q]
        A3[q]      = A3[q] :+ gSdiff0[q]
    }

    return(A1, A2, A3)
}

```

```

// Score function: Ancillary parameter
function stipw_gompertz_anc_score(struct stipw_info scalar S)
{
    real scalar loghr, loglambda, anc, gamma
    real matrix lp, ancp, apt, apS, apSdiff, q, apt0, apSO, apSdiff0, A1, A2, A3, A4

    loghr      = S.betas[1]
    loglambda  = S.betas[2]
    anc        = S.betas[3]
    gamma      = S.betas[4]

    lp          = exp(loglambda :+ loghr :* S.trt)
    ancp       = gamma :+ S.trt .* anc
    apt         = ancp :* S.t
    apS        = lp :* ancp:^(-1) :* (exp(apt) :- 1)
    apSdiff    = lp :* ancp:^(-2) :* (exp(apt) :* (apt :- 1) :+ 1)

    A1          = S.trt :* (S.d :- apS)
    A2          = S.d :- apS
    A3          = S.trt :* (S.d :* S.t :- apSdiff)
    A4          = S.d:*S.t :- apSdiff

    q           = selectindex(S.t0)
    if (rows(q) > 0) {
        apt0      = ancp :* S.t0
        apSO      = lp :* ancp:^(-1) :* (exp(apt0) :- 1)
    }
}

```

```

apSdiff0 = lp :* ancp:^(-2) :* (exp(apt0) :* (apt0 :- 1) :+ 1)

A1[q]   = A1[q] :+ S.trt[q] :* apS0[q]
A2[q]   = A2[q] :+ apS0[q]
A3[q]   = A3[q] :+ S.trt[q] :* apSdiff0[q]
A4[q]   = A4[q] :+ apSdiff0[q]
}

return(A1, A2, A3, A4)
}

```

```

///////////
// streg - Log-logistic //
///////////

```

```

// Score function: No ancillary
function stipw_loglogistic_score(struct stipw_info scalar S)
{
    real scalar logtr, loglambda, eg
    real matrix lp, u, s, q, u0, s0, A1, A2, A3

    logtr      = S.betas[1]
    loglambda  = S.betas[2]
    eg         = exp(-S.betas[3])

    lp          = loglambda :+ logtr :* S.trt
    u           = exp(-eg :* lp) :* S.t :^ eg
    s           = (S.d :+ 1) :* eg :/ (1 :+ u :^ (-1))

    A1          = S.trt :* (-1 :* S.d :* eg :+ s)
    A2          = -1 :* S.d :* eg :+ s
    A3          = -1 :* S.d :* eg :* (-1 :* lp :+ log(S.t)) - S.d :+ s :* (-1 :* lp :+ log(S.t))

    q           = selectindex(S.t0)
    if (rows(q) > 0) {
        u0        = exp(-eg :* lp) :* S.t0 :^ eg
        s0        = eg :/ (1 :+ u0 :^ (-1))

        A1[q]    = A1[q] :- S.trt[q] :* s0[q]
        A2[q]    = A2[q] :- s0[q]
        A3[q]    = A3[q] :- s0[q] :* (-1 :* lp[q] :+ log(S.t0[q]))
    }

    return(A1, A2, A3)
}

```

```

// Score function: Ancillary
function stipw_loglogistic_anc_score(struct stipw_info scalar S)
{
    real scalar logtr, loglambda, loganc, loggamma
    real matrix lp, ancp, u, s, q, u0, s0, A1, A2, A3, A4

    logtr      = S.betas[1]
    loglambda  = S.betas[2]
    loganc     = S.betas[3]
    loggamma   = S.betas[4]

    lp          = loglambda :+ logtr :* S.trt
    ancp       = exp(-loggamma :- loganc :* S.trt)

```

```

u      = exp(-ancp :* lp) :* S.t :^ ancp
s      = (S.d :+ 1) :* ancp :/ (1 :+ u :^ (-1))

A1     = S.trt :* (-1 :* S.d :* ancp :+ s)
A2     = -1 :* S.d :* ancp :+ s
A3     = S.trt :* (-1 :* S.d :* ancp :* (-1 :* lp :+ log(S.t)) - S.d :+ s :* (-1 :* lp :+ log(S.t)))
A4     = -1 :* S.d :* ancp :* (-1 :* lp :+ log(S.t)) - S.d :+ s :* (-1 :* lp :+ log(S.t))

q      = selectindex(S.t0)
if (rows(q) > 0) {
    u0    = exp(-ancp :* lp) :* S.t0 :^ ancp
    s0    = ancp :/ (1 :+ u0 :^ (-1))

    A1[q] = A1[q] :- S.trt[q] :* s0[q]
    A2[q] = A2[q] :- s0[q]
    A3[q] = A3[q] :- S.trt[q] :* s0[q] :* (-1 :* lp[q] :+ log(S.t0[q]))
    A4[q] = A4[q] :- s0[q] :* (-1 :* lp[q] :+ log(S.t0[q]))
}

return(A1, A2, A3, A4)
}

```

```

///////////////
// streg - Log normal //
/////////////

```

```

// Score function: No ancillary
function stipw_lognormal_score(struct stipw_info scalar S)
{
    real scalar logtr, mu, sd
    real matrix lp, u, du, pu, f, s
    real matrix q, lp0, u0, du0, pu0, s0
    real matrix A1, A2, A3

    logtr      = S.betas[1]
    mu         = S.betas[2]
    sd         = exp(S.betas[3])

    lp          = log(S.t) :- mu :- logtr :* S.trt
    u           = lp :/ sd
    du          = normalden(u)
    pu          = normal(u)
    f           = S.d :/ (sd^2) :* lp
    s           = (1 :- S.d) :* du :/ (sd :* (1 :- pu))

    A1          = S.trt :* (f :+ s)
    A2          = f :+ s
    A3          = -1 :* S.d :+ f :* lp :+ (1 :- S.d) :* du :* u :/ (1 :- pu)

    q           = selectindex(S.t0)
    if (rows(q) > 0) {
        lp0      = log(S.t0) :- mu :- logtr :* S.trt
        u0       = lp0 :/ sd
        du0      = normalden(u0)
        pu0      = normal(u0)
        s0       = du0 :/ (sd :* (1 :- pu0))

        A1[q]   = A1[q] :- S.trt[q] :* s0[q]
        A2[q]   = A2[q] :- s0[q]
        A3[q]   = A3[q] :- du0[q] :* u0[q] :/ (1 :- pu0[q])
    }
}

```

```

}

return(A1, A2, A3)
}

// Score function: Ancillary
function stipw_lognormal_anc_score(struct stipw_info scalar S)
{
    real scalar logtr, mu, sdcons, sdanc
    real matrix lp, sd, u, du, pu, f, s
    real matrix q, lp0, u0, du0, pu0, s0
    real matrix A1, A2, A3, A4

    logtr      = S.betas[1]
    mu         = S.betas[2]
    sdanc      = S.betas[3]
    sdcons     = S.betas[4]

    lp          = log(S.t) :- mu :- logtr :* S.trt
    sd          = exp(sdcons :+ sdanc :* S.trt)
    u           = lp :/ sd
    du          = normalden(u)
    pu          = normal(u)
    f            = S.d :/ (sd:^2) :* lp
    s            = (1 :- S.d) :* du :/ (sd :* (1 :- pu))

    A1          = S.trt :* (f :+ s)
    A2          = f :+ s
    A3          = S.trt :* (-1 :* S.d :+ f :* lp :+ (1 :- S.d) :* du :* u :/ (1 :- pu))
    A4          = -1 :* S.d :+ f :* lp :+ (1 :- S.d) :* du :* u :/ (1 :- pu)

    q           = selectindex(S.t0)
    if (rows(q) > 0) {
        lp0        = log(S.t0) :- mu :- logtr :* S.trt
        u0         = lp0 :/ sd
        du0        = normalden(u0)
        pu0        = normal(u0)
        s0          = du0 :/ (sd :* (1 :- pu0))

        A1[q]      = A1[q] :- S.trt[q] :* s0[q]
        A2[q]      = A2[q] :- s0[q]
        A3[q]      = A3[q] :- S.trt[q] :* du0[q] :* u0[q] :/ (1 :- pu0[q])
        A4[q]      = A4[q] :- du0[q] :* u0[q] :/ (1 :- pu0[q])
    }

    return(A1, A2, A3, A4)
}

```

```

///////////
// stpm2 - Hazard model //
/////////

```

```

// Score function
function stipw_stpm2_hazard_score(struct stipw_info scalar S)
{
    real matrix g1, g2, A, q

    g1          = S.d :- exp(S.xb)
    g2          = S.d :/ S.dxb

```

```
if (S.tvc ==1) A = g1 :* (S.trt,S.rcs,S.rcs_trt,J(S.n,1,1)) + g2 :* (J(S.n,1,0),S.drcs,S.drcs_trt,J(S.n,1,0))
else          A = g1 :* (S.trt,S.rcs,J(S.n,1,1))+ g2 :* (J(S.n,1,0),S.drcs,J(S.n,1,0))

q           = selectindex(S.t0)
if (rows(q) > 0) {                                // delayed entry
    if (S.tvc ==1)      A[q,] := exp(S.xb0)[q] :* (S.trt[q],S.s0_rcs[q],S.s0_rcs_trt[q],J(rows(q),1,1))
    else              A[q,] := A[q,] + exp(S.xb0)[q] :* (S.trt[q],S.s0_rcs[q],J(rows(q),1,1))
}

return(A)
}

end
```

# Appendix D

---

## Code for `msaj`

---

The user-written command `msaj`, part of the `multistate` package [22], in Stata is available on the SSC archive and can be accessed from GitHub at <https://github.com/RedDoorAnalytics/multistate/tree/main/msaj>. This command was majorly redeveloped and extended as part of the work of this thesis and gave rise to `msaj`, version 1.0.0. A copy of the code, version 1.0.0, dated 02.03.2020, is given here. Note that version 1.0.1, dated 11.09.2020, was used in Chapter 7. The only difference from version 1.0.0 to version 1.0.1 is that the option `enter` was renamed to `ltruncate`. `msaj` can be used to obtain a comprehensive set of non-parametric Aalen-Johansen estimates from a multistate model, including length of stay.

\*! Version 1.0.0 02Mar2020 MH

```
/*
History
MH 02mar2020: version 1.0.0 - restructured so calculations are all performed in mata (improved efficiency)
- bug fix: confidence intervals caused a conformability error and kronecker delta function
incorrectly defined; now fixed
    - removal of gen option
    - id option no longer required due to restructure
    - enter time option added
    - exit time option added
    - from state option added
    - los option added
    - se option added
    - program now works for bi-directional models
    - additional error checks
```

PCL 30apr2019: version 0.6.0 -
\*/

```
program define msaj
    version 14.2
    syntax [if] [in] , [ TRANSMATRIX(name) ///
        BY(varname)      ///
        ENTER(real 0)    ///
        EXIT(real -99)   ///
        FROM(integer 1)  ///
        CR               ///
        LOS              ///
        CI               ///
        SE               ///
    ]
    marksample touse
    if "`by'" != "" {
        qui replace `touse' = 0 if `by' == .
        qui levelsof `by' if `touse', local(bylevels)
    }
    // Error checks
    cap confirm variable _t _d _st _t0, exact
    if _rc {
        di as error "Data must be stset (at least one variable of _st, _t, _t0 or _d missing)"
        exit 198
    }
    cap confirm variable _trans, exact
    if _rc {
        di as error "Data must be misset (_trans variable missing)"
        exit 198
    }
    cap confirm variable _to, exact
    if _rc & "cr"!="{
        di as error "Data must be misset (_to variable missing, needed for CR option)"
        exit 198
    }
    cap confirm variable _to _from, exact
    if ("ci" != "" | "se" != "") & _rc {
```

```

di as error "_from and _to variables must be specified for CI or SE option"
exit 198
}

if ("`transmatrix'" == "" & "`cr'" == "") | ("`transmatrix'" != "" & "`cr'" != "") {
    di as error "You must specify either the transition matrix or use the cr option"
    exit 198
}

if "`transmatrix'" != "" {
    local Nstates = colsof(`transmatrix')

    if `Nstates' < 2 {
        di as error "Must be at least 2 possible states, including starting state"
        exit 198
    }

    if `from' < 1 | `from' > `Nstates' {
        di as error "From state must be between 1 and the maximum number of states"
        exit 198
    }
}

if "`cr'" != "" & `from'!=1 {
    di as error "If CR is specified then from must be 1"
    exit 198
}

if `enter' < 0 {
    di as error "Enter time must be positive"
    exit 198
}

// Set exit time if not given and check it is after enter time

summ _t if `touse', meanonly
local max_t = `r(max)'
summ _t if `touse' & _d ==1, meanonly
local max_event = `r(max)'

if `exit' == -99 local exit `max_event'

if `exit' <= `enter' {
    di as error "Enter time (default 0) is greater than or equal to exit time (default max any event time)"
    exit 198
}

if `exit' > `max_event' & `exit' <= `max_t' {
    di as error "Warning: Exit time is greater than last event time"
}

if `exit' > `max_t' {
    di as error "Exit time is greater than the maximum time"
    exit 198
}

// Build the transition matrix if CR is specified

if "`cr'" != "" {

```

```

summ _to if `touse', meanonly
local Nstates `r(max)'

if `Nstates' < 2 {
    di as error "Must be at least 2 possible states, including starting state (max _to is 1)"
    exit 198
}

tempname transmatrix
matrix `transmatrix' = J(`Nstates',`Nstates',.)
forvalues i = 2/`Nstates' {
    local tmptrans = `i' - 1
    matrix `transmatrix'[1,`i']= `tmptrans'
}
}

// Prepare the output variables

forvalues i = 1/`Nstates' {
    local newvars `newvars' P_AJ_`i'
    if "`ci'" != "" {
        local newvars `newvars' P_AJ_`i'_lci P_AJ_`i'_uci
    }
    if "`se'" != "" {
        local newvars_se `newvars_se' P_AJ_`i'_se
    }
    if "`los'" != "" {
        local newvars_LOS `newvars_LOS' LOS_AJ_`i'
    }
}
}

local Nnewvars = wordcount("`newvars'")
forvalues i = 1/`Nnewvars' {
    local tmp = word("`newvars'",`i')
    qui gen double `tmp'=.
}

if "`se'" != "" {
    local Nnewvars_se = wordcount("`newvars_se'")
    forvalues i = 1/`Nnewvars_se' {
        local tmp = word("`newvars_se'",`i')
        qui gen double `tmp'=.
    }
}

if "`los'" != "" {
    local Nnewvars_LOS = wordcount("`newvars_LOS'")
    forvalues i = 1/`Nnewvars_LOS' {
        local tmp = word("`newvars_LOS'",`i')
        qui gen double `tmp'=.
    }
}

// Get the hazards

tempvar Nrisk Nevents

sts gen `Nrisk'=n if `touse', by(_trans `by')

```

```

sts gen `Nevents'=d if `touse', by(_trans `by')

// Call mata to apply AJ equations
mata AJ()

end

mata:

// Main program
void function AJ()
{

    // Import data into mata
    touse = st_local("touse")
    st_view(t,,"_t",touse)
    st_view(d,,"_d",touse)
    st_view(trans,,,"_trans",touse)
    st_view(Nrisk,..st_local("Nrisk"), touse)
    st_view(Nevents,..st_local("Nevents"), touse)

    // Read in by data
    if(st_local("by") != "") {
        bylevels = strtoreal(tokens(st_local("bylevels")))
    }
    else {
        bylevels = 1
        by = J(rows(t),1,1)
    }

    // Get unique rows for time, trans, haz, from, to, Nrisk and Nevents (only at event times)
    haz = Nevents:/Nrisk
    d_index = selectindex(d==1)
    data_all = uniqrows((t[d_index], by[d_index], trans[d_index], haz[d_index]))

    // Get ci information
    ci = st_local("ci") != ""
    se = st_local("se") != ""
    if (ci | se) {
        from = st_view(from,,,"_from",touse)
        to = st_view(to,,,"_to",touse)
        dataVar_all = uniqrows((t[d_index], by[d_index], from[d_index], to[d_index],
        Nrisk[d_index], Nevents[d_index]))
    }

    // Get transmatrix, states, transitions
    transmat = st_matrix(st_local("transmatrix"))
    transmat_index = transRowCol(transmat)
    Ntrans = check_transmatrix(transmat)
    Nstates = rows(transmat)

    // Get enter and exit time
    entertime = strtoreal(st_local("enter"))
    exittime = strtoreal(st_local("exit"))

    // Get from state & check not absorbing
    fromS = strtoreal(st_local("from"))
    if (max(transmat[fromS,]) == .) {
        errprintf("From state is an absorbing state\n")
    }
}

```

```

    exit(198)
}

// Get information for LOS
los           = st_local("los") != ""

// Create variables to store results
if (ci) P_state          = J(rows(t), Nstates*3,.)
else P_state             = J(rows(t), Nstates,.)
if (se) se_state          = J(rows(t), Nstates,.)
if (los) LOS_state        = J(rows(t), Nstates,.)

// Get useful matrices
Imat           = I(Nstates)
if (ci | se) zeros       = J(Nstates, Nstates, 0)

// Main loop for calculations, loop over by
for(k=1;k<=cols(bylevels);k++) {

    b               = bylevels[k]

    // Set initial probabilities
    P               = I(Nstates)
    if (ci | se) varP      = J(Nstates:^2, Nstates:^2, 0)

    // Select data for by level
    by_d_index      = selectindex(data_all[,2]==b)
    data            = data_all[by_d_index,]
    if (ci | se) dataVar     = dataVar_all[by_d_index,]
    t_d_unique      = uniqrows(data[,1])
    Ntd             = rows(t_d_unique)

    if (los) {
        by_index      = selectindex(by==b)
        t_unique      = uniqrows(t[by_index])
        Nt             = rows(t_unique)
        P_unique      = J(Nt, Nstates,.)
        P_t_unique    = J(Nt, 1, .)
        v              = 1
        past_t         = -1
        LOS            = J(1, Nstates, 0)
        t0             = entertime
        P1             = J(Nt, Nstates, .)
        l              = 1
    }

    // Work out the probabilities
    for(u=1; u<=Ntd; u++) {

        // Get current t and check before exit time and after enter time
        current_t      = t_d_unique[u]
        if (current_t < entertime | current_t > exittime) continue

        // If current t = enter time then this will be first current t and P=identity already

        // If current t > enter time then work out the probability
        if (current_t > entertime) {

            // Build hazard matrix for current time
            H             = J(Nstates, Nstates, 0)
        }
    }
}

```

```

// Fill in the off-diagonals
for(j=1;j<=Ntrans;j++) {

    if(max((data[,1]==current_t) & (data[,3]==j))==1) {
        H[transmat_index[j,1],transmat_index[j,2]] =
select(data[,4],((data[,1]==current_t) & (data[,3]==j)) ==1)
    }
}

// Work out the diagonals
for(i=1;i<=Nstates;i++) {
    H[i,i]=-quadsum(H[,i])
}

// Work out variance
if (ci | se) {

    // Work out Y_k and N_k., N_kn and N_kl before main loop for efficiency
    RiskEvents      = select(dataVar,dataVar[,1]==current_t)
    atrisk_k        = J(1, Nstates, 0)
    events_k        = J(1, Nstates, 0)
    events_kn       = J(Nstates, Nstates, 0)
    events_kl       = J(Nstates, Nstates, 0)

    for(vk = 1;vk<=Nstates;vk++) {

        any_atrisk_k = select(RiskEvents[,5],RiskEvents[,3]==vk)
        if (rows(any_atrisk_k) != 0) {

            atrisk_k[vk] = any_atrisk_k[1]
            events_k[vk] = sum(select(RiskEvents[,6],RiskEvents[,3]==vk))

            for(vl = 1;vl<=Nstates;vl++) {
                any_events_kl = select(RiskEvents[,6],RiskEvents[,3]==vk
:& RiskEvents[,4]==vl)
                if(rows(any_events_kl) != 0) events_kl[vk, vl] =
any_events_kl
            }

            for(vn = 1;vn<=Nstates;vn++) {
                any_events_kn =
select(RiskEvents[,6],RiskEvents[,3]==vk & RiskEvents[,4]==vn)
                if(rows(any_events_kn) != 0) events_kn[vk, vn] =
any_events_kn
            }
        }
        atrisk_k3 = atrisk_k^(-3)

        // Calculate the covariance matrix of (A_kl, A_mn)
        // Stored in (l,n)th block with row k and column m
        VarBlock = asarray_create("real",2)
        for(vl = 1;vl<=Nstates;vl++) {

            for(vn = 1;vn<=Nstates;vn++) {

                tempBlock = zeros
                delta_ln = (vl == vn)

```

```

        for(vk = 1;vk<=Nstates;vk++) {

            if (atrisk_k[vk] != 0) {

                if(vk == vl & vk == vn & events_k[vk] != 0) {
                    tempBlock[vk,vk] = (atrisk_k[vk] -
events_k[vk])*events_k[vk]*atrisk_k3[vk]
                }
            }

            if(vk == vl & vk != vn & events_kn[vk,vn] != 0) {
                tempBlock[vk,vk] = -(atrisk_k[vk] -
events_k[vk])*events_kn[vk,vn]*atrisk_k3[vk]
            }

            if(vk != vl & vk != vn & events_kn[vk,vn] != 0) {
                tempBlock[vk,vk] =
(delta_ln*atrisk_k[vk] - events_kl[vk,vl])*events_kn[vk,vn]*atrisk_k3[vk]
            }
        }

        asarray(VarBlock,(vl,vn),tempBlock)

    }

}

// Format covariance to be a huge martix rather than array
VarAd = J(Nstates:^2,Nstates:^2,0)
for(vl = 1;vl<=Nstates;vl++) {
    for(vn = 1;vn<=Nstates;vn++) {
        trows = ((vl-1)*Nstates+1)..((vl-1)*Nstates+Nstates)
        tcols = ((vn-1)*Nstates+1)..((vn-1)*Nstates+Nstates)
        VarAd[trows,tcols] = asarray(VarBlock,(vl,vn))
    }
}

// Make sure it is symmetrical
for(i = 1; i<= Nstates:^2; i++) {
    for (j = 1; j<= Nstates:^2; j++) {
        if (VarAd[i,j] == 0 & VarAd[j,i] != 0) VarAd[i,j] = VarAd[j,i]
    }
}

// Calculate the variance
tmp1      = ((Imat + H)' # Imat) * varP * ((Imat + H) # Imat)
tmp2      = (Imat # P) * VarAd * (Imat # P')
varP      = tmp1 + tmp2
}

// Matrix multiplication - calculate probability
P = P*(Imat + H)
}

// Store results
res_index = selectindex(t==current_t :& d==1 :& by==b)

if (ci | se) {
    seP = (rowshape(sqrt(diagonal(varP)),Nstates)[,fromS])'
}

```

```

if (se) se_state[res_index,] = J(rows(res_index),1,seP[1,])
if (ci) {
    for(j=1;j<=Nstates;j++) {
        P_colindex = (3*(j-1)+1)
        P_state[res_index,P_colindex..P_colindex+2] =
J(rows(res_index),1,(P[fromS,j], P[fromS,j] :-1.96:*seP[1,j], P[fromS,j] :+1.96:*seP[1,j]))
    }
}
if (ci != 1) P_state[res_index,] = J(rows(res_index),1,P[fromS,])

if (los) {
    if (current_t != past_t) {
        P_t_unique[v] = current_t
        P_unique[v,] = P[fromS,]
        v = v + 1
    }
    past_t = current_t
}
}

// LOS
if (los) {

    for(u=1; u<=Nt; u++) {

        // Get current t and check before exit time and after enter time
        current_t = t_unique[u]
        if (current_t < entertime | current_t > exittime) continue

        // Work out expanded prob matrix
        if (P_t_unique[l] != .) {
            if (current_t == P_t_unique[l]) {
                P1[u,] = P_unique[l,]
                l = l + 1
            }
            else {
                if (l == 1) P1[u, ] = Iimat[fromS,]
                else P1[u, ] = P_unique[l-1,]
            }
        }
        else {
            if (l == 1) P1[u, ] = Iimat[fromS,]
            else P1[u, ] = P_unique[l-1,]
        }

        // Calculate LOS
        if (t0 == entertime) LOS = LOS :+ (current_t - t0):*Iimat[fromS,]
        else LOS = LOS :+ (current_t - t0):*P1[u-1,]

        // Store LOS
        los_index = selectindex(t==current_t :& by==b)
        LOS_state[los_index,] = J(rows(los_index),1,LOS)

        // Get t0 for next iteration
        t0 = current_t
    }
}
}

```

```

// Check all probabilities within [0,1]
P_state = 0:*(P_state:<0) :+ 1:*(P_state:>1) :+ P_state:*(P_state:>=0 & P_state:<=1)

// Output results
newvars = tokens(st_local("newvars"))
st_store(.,newvars,touse,P_state)

if (se) {
    newvars_se = tokens(st_local("newvars_se"))
    st_store(.,newvars_se,touse,se_state)
}

if (los) {
    newvars_LOS = tokens(st_local("newvars_LOS"))
    st_store(.,newvars_LOS,touse,LOS_state)
}
}

// Does checks and returns No. of transitions
function check_transmatrix(tmat)
{
    tmat_ind = tmat:!=.
    if (max(diagonal(tmat_ind))>0) {
        errprintf("All elements on the diagonal of transmatrix() must be coded missing = .\n")
        exit(198)
    }

    row = 1
    rtmat = rows(tmat)
    trans = 1
    while (row<rtmat) {
        for (i=1;i<=rtmat;i++) {

            if (sum(tmat==tmat[row,i])>1 & tmat[row,i]!=.) {
                errprintf("Elements of transmatrix() are not unique\n")
                exit(198)
            }

            if (tmat[row,i]!=. & tmat[row,i]!=trans){
                errprintf("Elements of transmatrix() must be sequentially numbered from 1,...,K, where K =
number of transitions\n")
                exit(198)
            }

            if (tmat[row,i]!=.) trans++
        }

        row++
    }

    return(trans-1)
}

// Creates to/from states for each transition
function transRowCol(tmat)
{
    tmat_index = J(max(tmat),2,.)
    row = 1

```

```
rtmat = rows(tmat)
trans = 1

while (row<rtmat) {
    for (i=1;i<=rtmat;i++) {
        if(tmat[row,i] == trans) {
            tmat_index[trans,] = (row,i)
            trans++
        }
    }
    row++
}
return(tmat_index)
}
end
```

# Appendix E

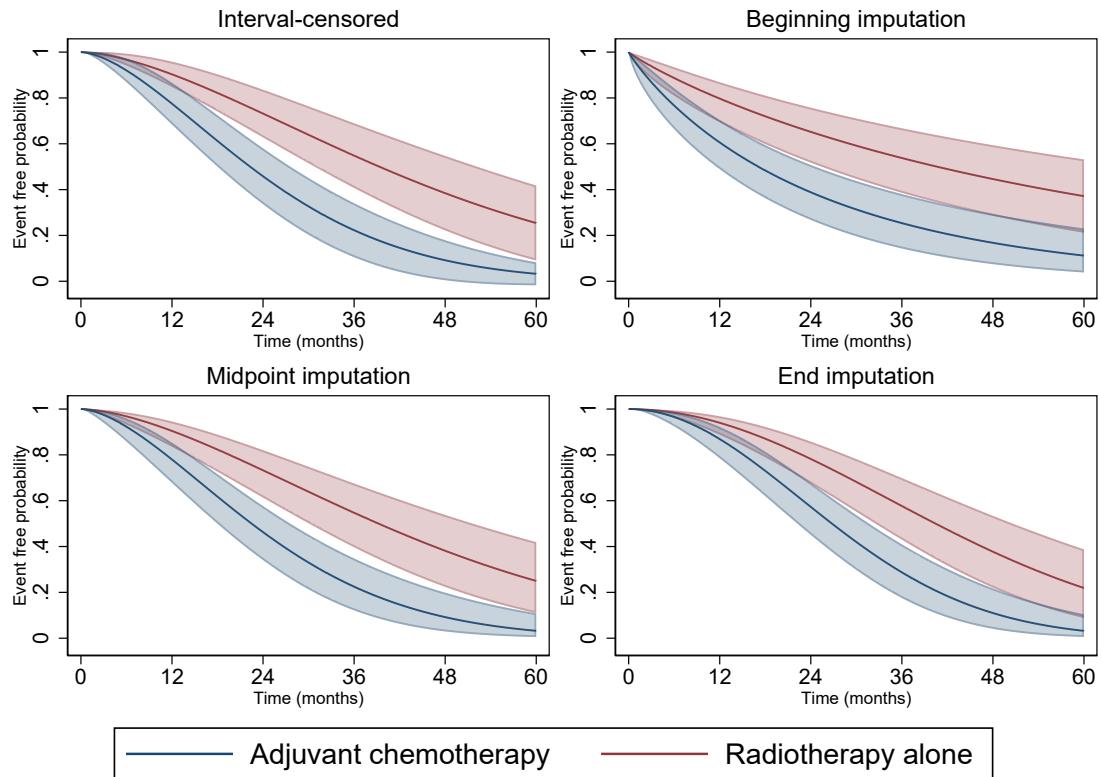
---

## Additional Material for Chapter 3

---

This appendix provides additional figures and tables for Chapter 3: The impact of naive imputation on interval-censored survival data. An additional figure is given for the analysis of the motivating dataset performed in Section 3.4. Figures from the simulation study in Section 3.6.2 for the survival probability at 24, 36 and 48 months are then shown. This appendix finishes with tables from the simulation study in Section 3.6.2 for selected scenarios with the corresponding MCSE.

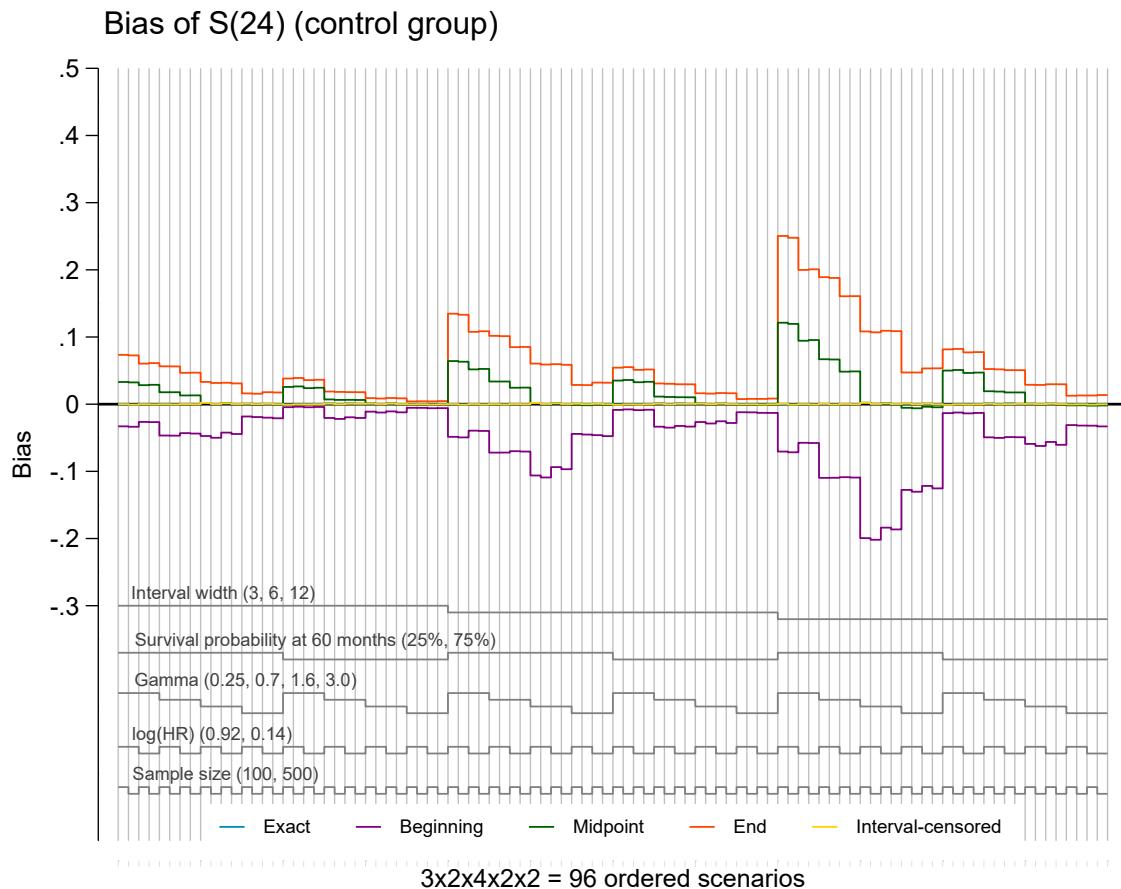
## E.1 Motivating Dataset



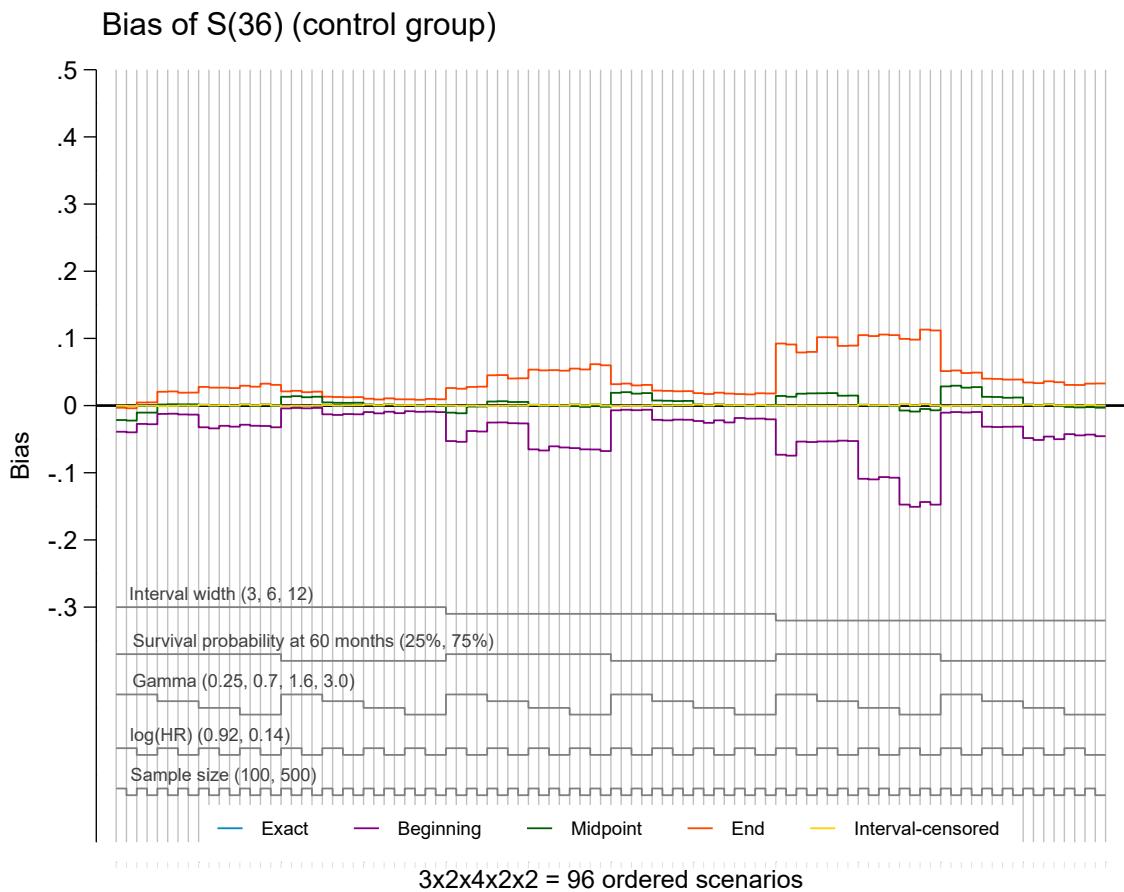
**Figure E.1:** Survival probability estimates and confidence intervals for the breast cosmesis data using the appropriate likelihood-based approach (top left), beginning imputation (top right), midpoint imputation (bottom left) and end imputation (bottom right)

## E.2 Simulation Study: Additional Figures for the Survival Probability at 24, 36 and 48 Months

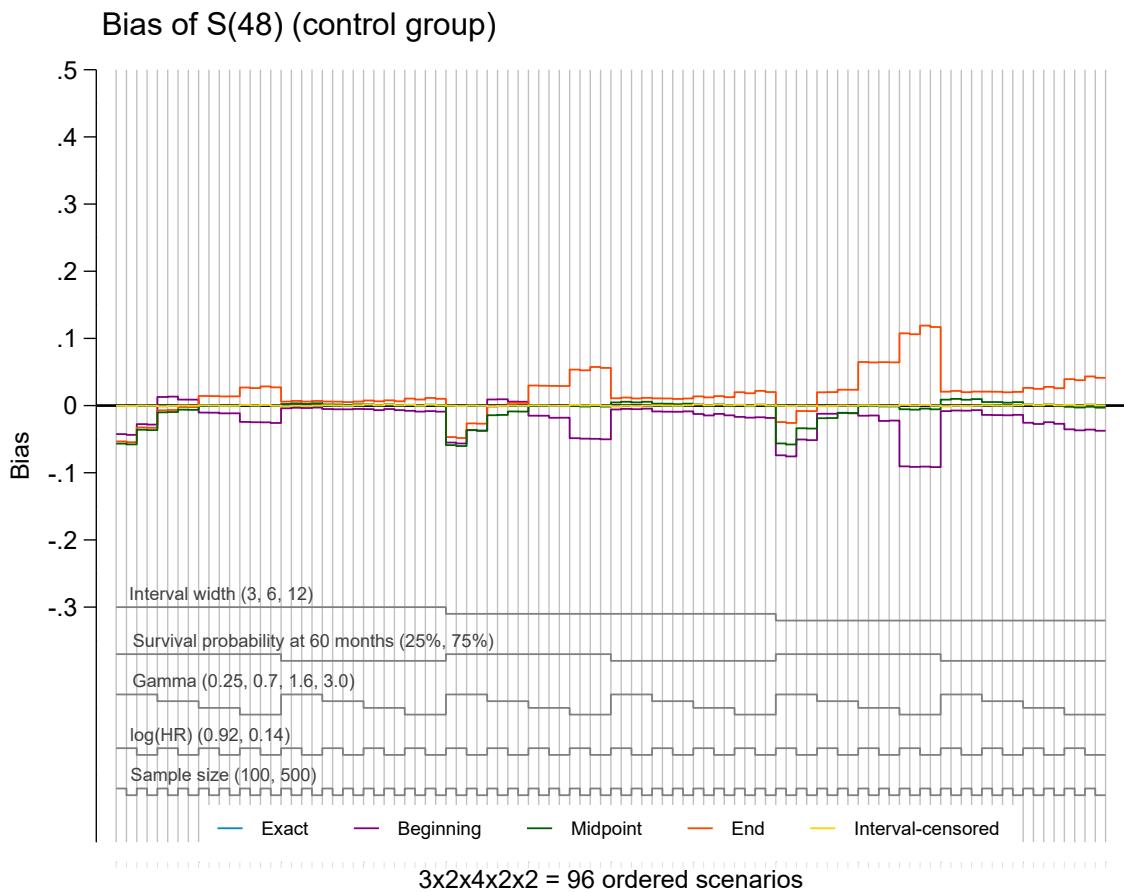
### E.2.1 Aim 1: Bias



**Figure E.2:** Nested loop plot showing the bias of the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

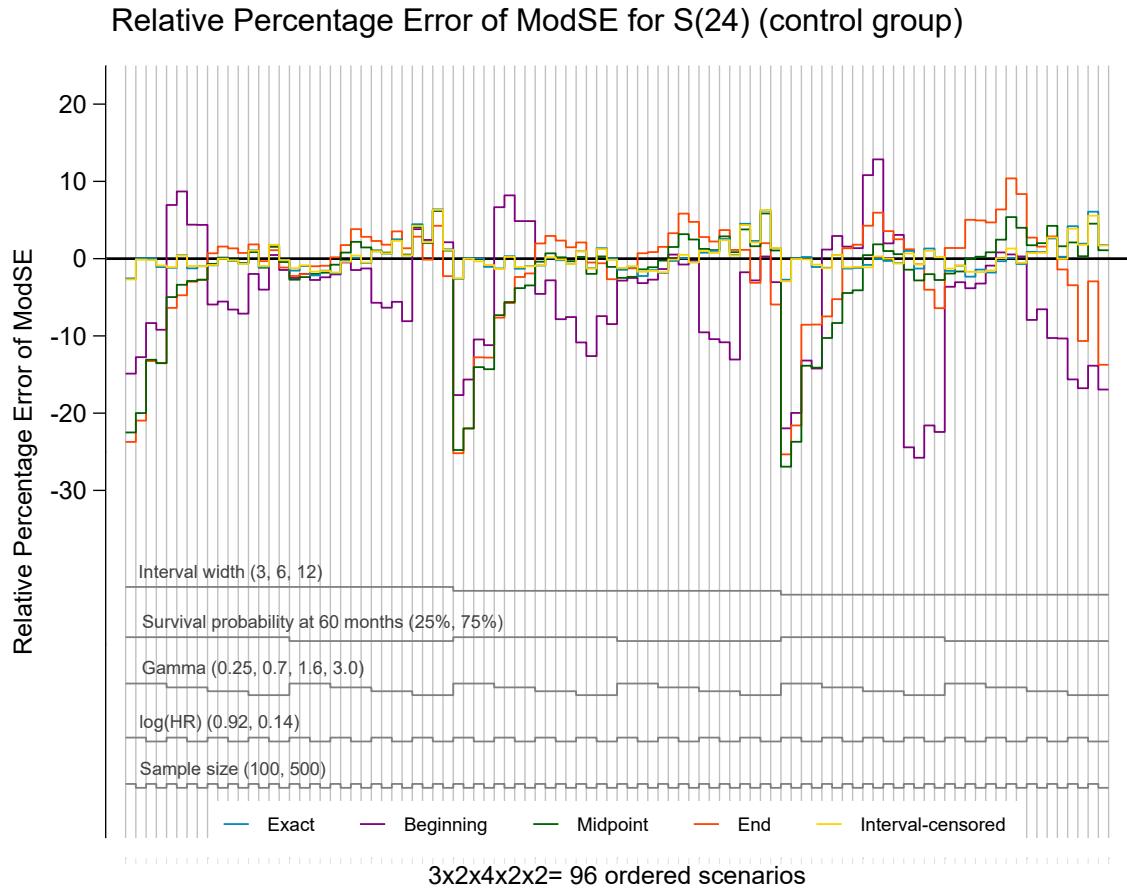


**Figure E.3:** Nested loop plot showing the bias of the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

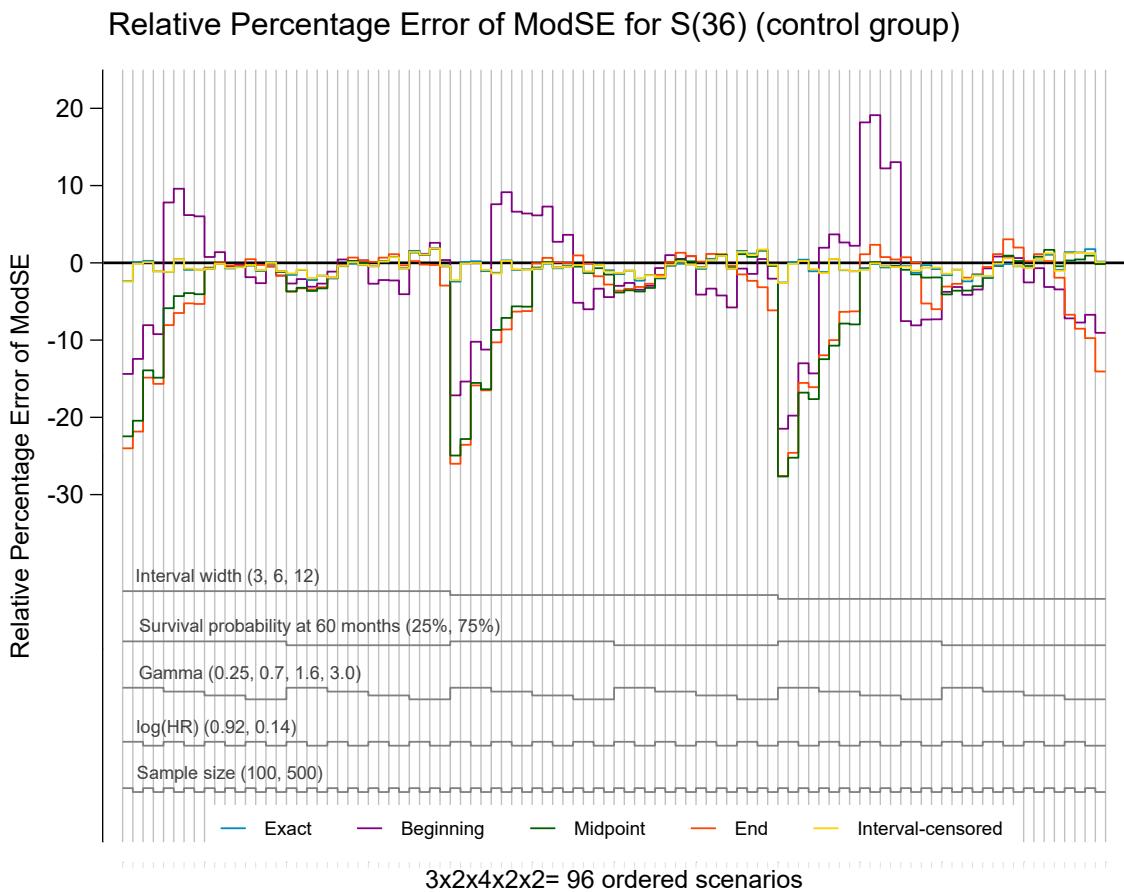


**Figure E.4:** Nested loop plot showing the bias of the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

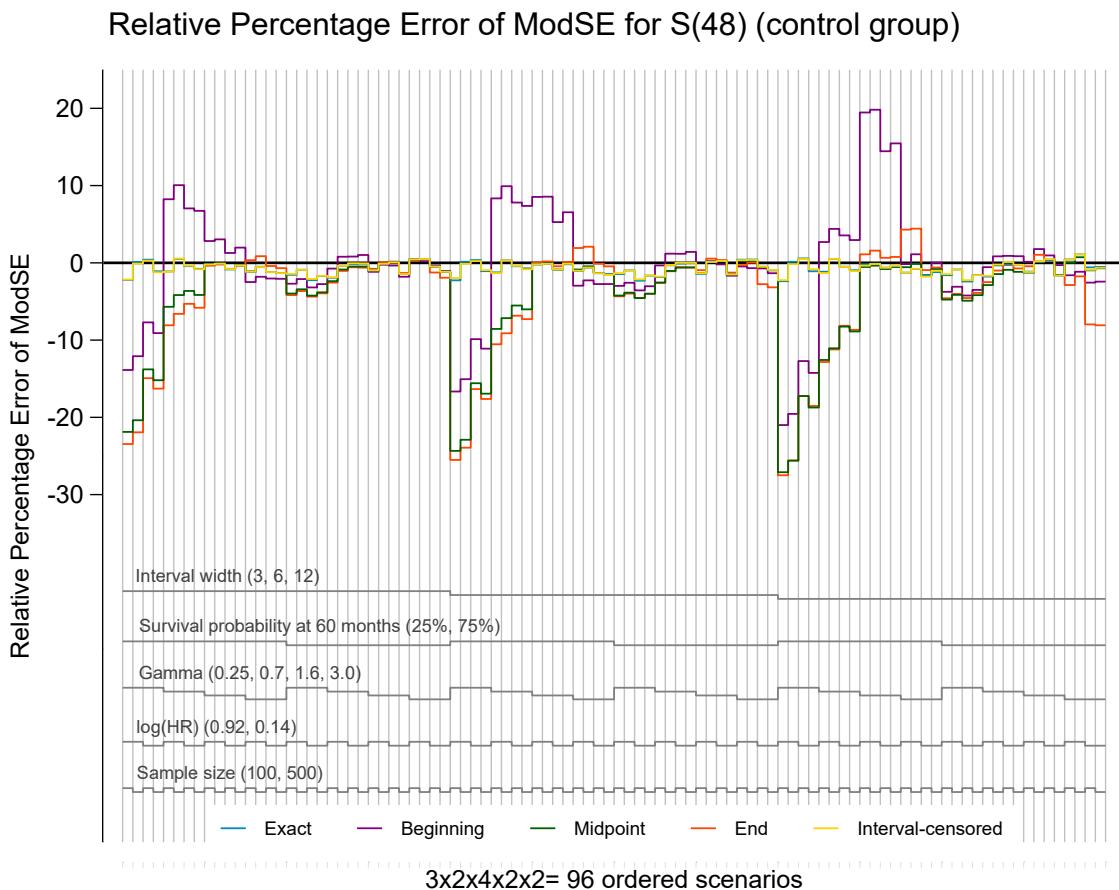
### E.2.2 Aim 2: Relative Percentage Error in ModSE



**Figure E.5:** Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

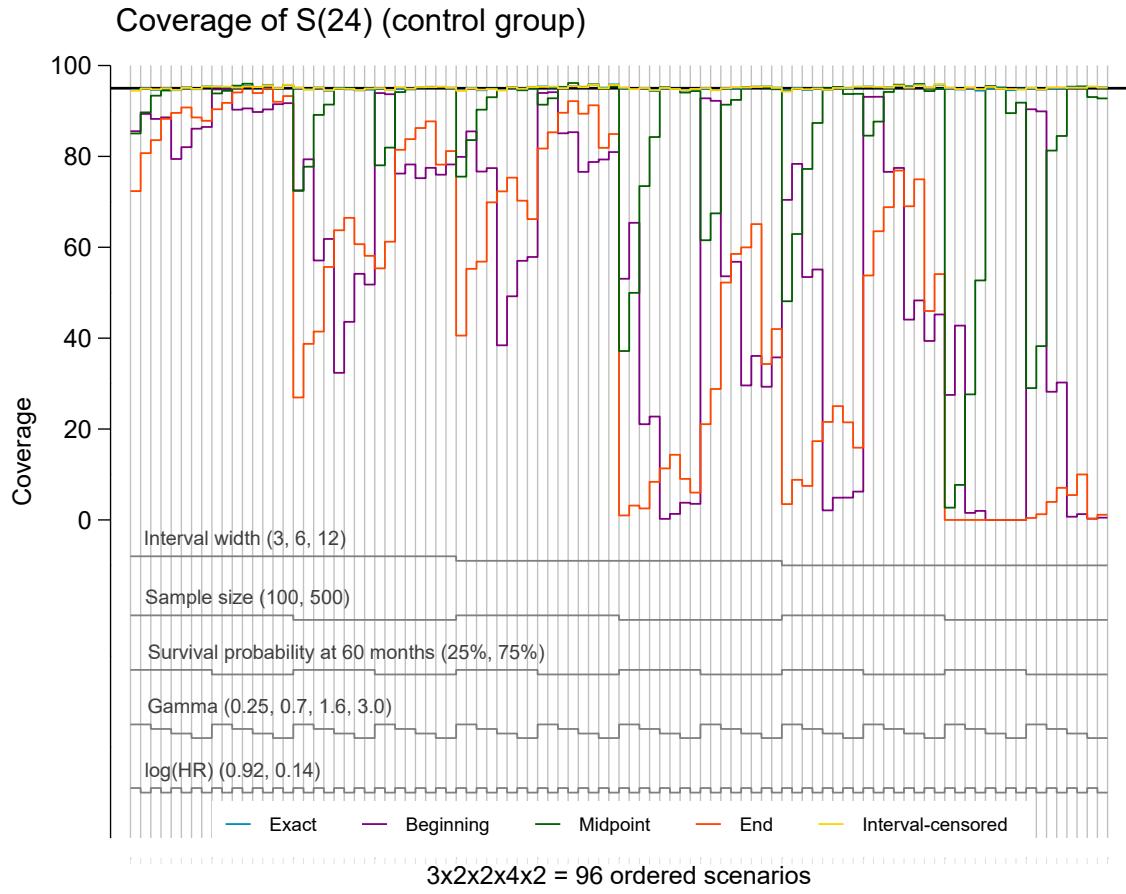


**Figure E.6:** Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

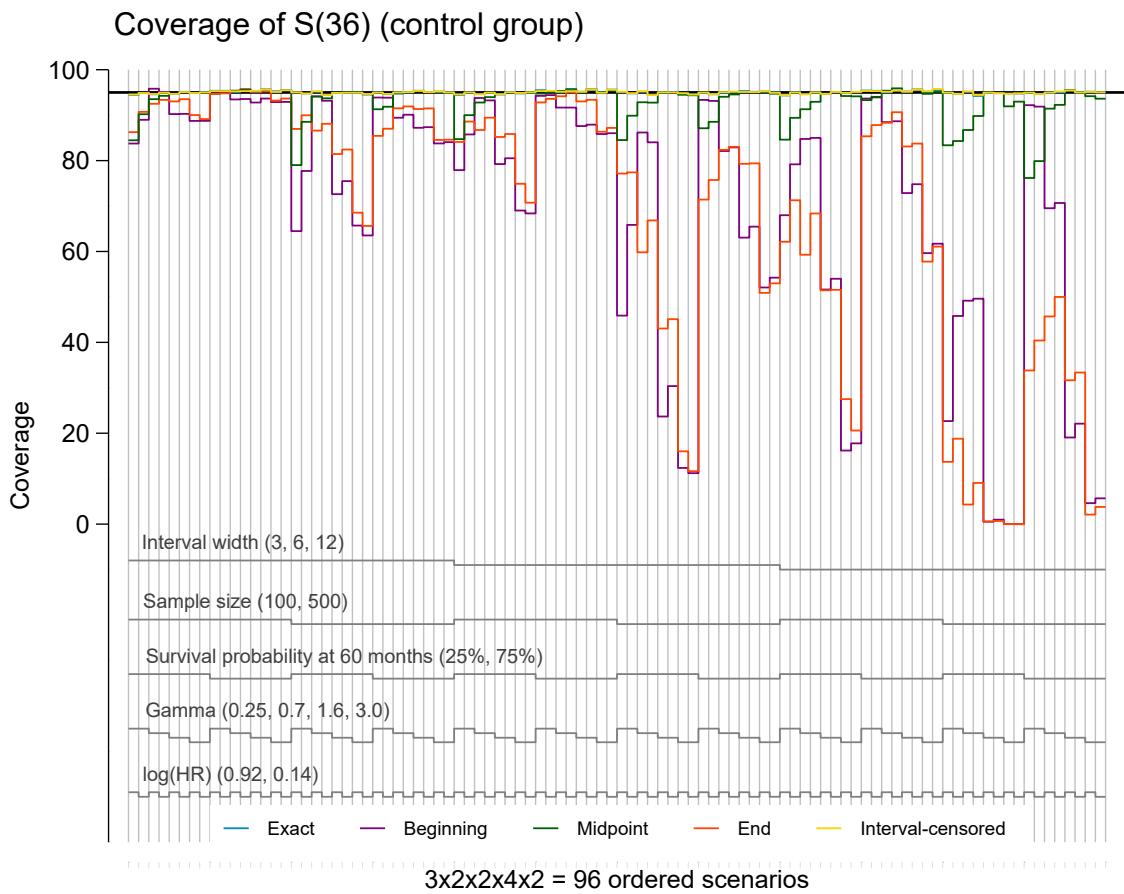


**Figure E.7:** Nested loop plot showing the relative percentage error in model-based standard errors for the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

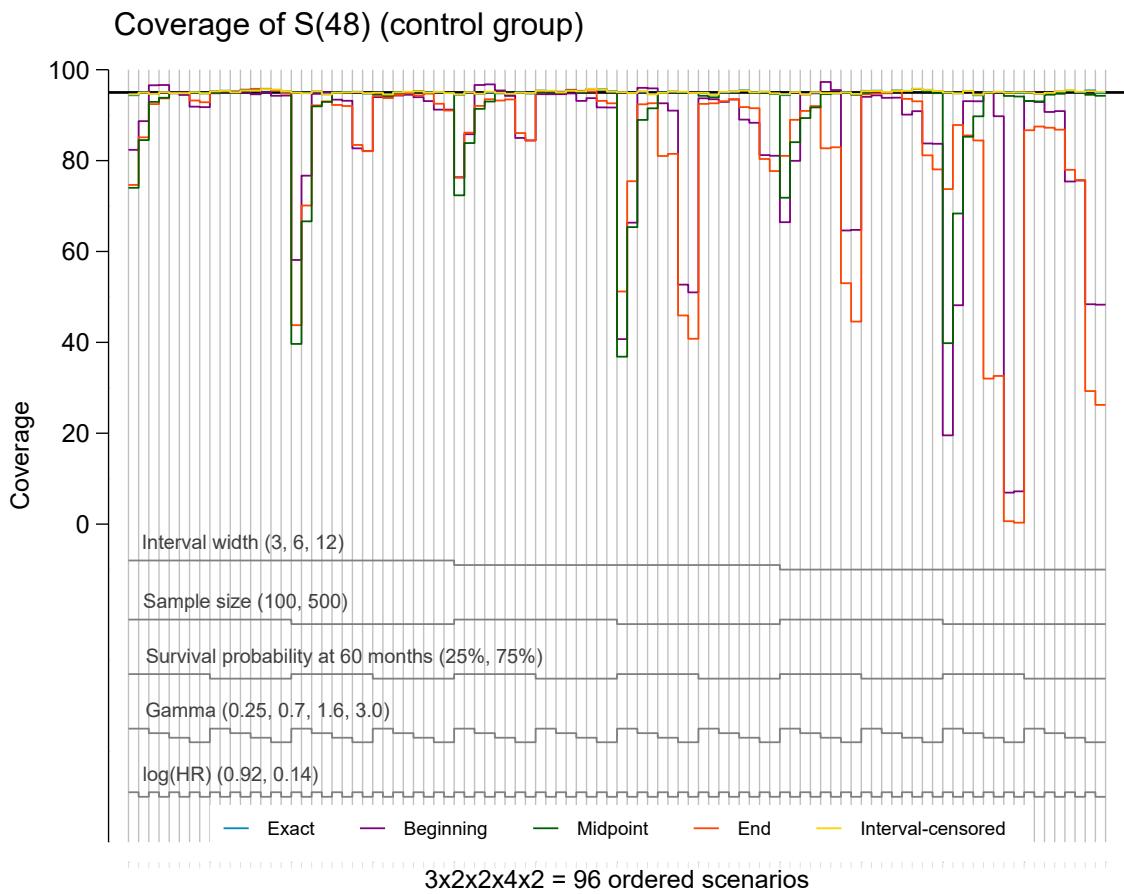
### E.2.3 Aim 3: Coverage



**Figure E.8:** Nested loop plot showing the coverage of the survival probability at 24 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method



**Figure E.9:** Nested loop plot showing the coverage of the survival probability at 36 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method



**Figure E.10:** Nested loop plot showing the coverage of the survival probability at 48 months (in the control group) across the 96 scenarios for each method. Each step represents a different scenario, which can be decoded using the key at the bottom of the graph. Note that the IC method often overlaps the exact method

## E.3 Simulation Study: Additional Tables

### E.3.1 Log Hazard Ratio

**Table E.1:** Bias with MCSE for the log hazard ratio for sample size 100 and interval width 12 months

$\beta$	$S_0(60)$	$\gamma$	Bias (MCSE)				
			Exact	Beg	Mid	End	IC
0.92	25%	0.25	0.0163 (0.0024)	0.2243 (0.0035)	0.2876 (0.0039)	0.2832 (0.0039)	0.0311 (0.0027)
		0.7	0.0199 (0.0024)	-0.0026 (0.0024)	0.1631 (0.0029)	0.1691 (0.0030)	0.0234 (0.0025)
		1.6	0.0221 (0.0024)	-0.2107 (0.0018)	0.0150 (0.0024)	0.0107 (0.0024)	0.0236 (0.0024)
		3	0.0211 (0.0024)	-0.0606 (0.0023)	-0.0123 (0.0023)	-0.0710 (0.0022)	0.0241 (0.0025)
	75%	0.25	0.0252 (0.0038)	0.0542 (0.0039)	0.0685 (0.0040)	0.0678 (0.0040)	0.0239 (0.0038)
		0.7	0.0241 (0.0038)	0.0066 (0.0037)	0.0425 (0.0039)	0.0396 (0.0039)	0.0234 (0.0038)
		1.6	0.0331 (0.0038)	0.0017 (0.0037)	0.0335 (0.0038)	0.0180 (0.0037)	0.0324 (0.0038)
		3	0.0219 (0.0038)	0.0284 (0.0038)	0.0228 (0.0038)	-0.0189 (0.0037)	0.0214 (0.0038)
0.14	25%	0.25	0.0019 (0.0024)	0.0180 (0.0028)	0.0238 (0.0029)	0.0236 (0.0029)	0.0024 (0.0025)
		0.7	0.0020 (0.0024)	-0.0037 (0.0024)	0.0127 (0.0026)	0.0124 (0.0026)	0.0021 (0.0025)
		1.6	0.0020 (0.0024)	-0.0184 (0.0021)	0.0015 (0.0024)	-0.0019 (0.0024)	0.0020 (0.0024)
		3	0.0039 (0.0024)	0.0009 (0.0024)	0.0020 (0.0024)	-0.0093 (0.0023)	0.0044 (0.0025)
	75%	0.25	0.0006 (0.0043)	0.0031 (0.0044)	0.0044 (0.0045)	0.0044 (0.0045)	0.0004 (0.0043)
		0.7	0.0079 (0.0042)	0.0062 (0.0042)	0.0097 (0.0043)	0.0092 (0.0042)	0.0078 (0.0042)
		1.6	0.0107 (0.0042)	0.0081 (0.0042)	0.0108 (0.0042)	0.0091 (0.0042)	0.0106 (0.0042)
		3	0.0035 (0.0043)	0.0044 (0.0043)	0.0040 (0.0043)	0.0001 (0.0042)	0.0038 (0.0043)

**Table E.2:** Relative percentage error of the model-based standard errors with MCSE for the log hazard ratio for sample size 100 and interval width 12 months

$\beta$	$S_0(60)$	$\gamma$	Relative % error in ModSE (MCSE)				
			Exact	Beg	Mid	End	IC
0.92	25%	0.25	-0.977 (0.733)	-29.793 (0.521)	-35.806 (0.478)	-36.230 (0.475)	-2.088 (0.727)
		0.7	-1.151 (0.730)	-4.143 (0.707)	-18.435 (0.602)	-20.354 (0.588)	-1.577 (0.727)
		1.6	-1.822 (0.725)	27.805 (0.943)	-1.720 (0.725)	-2.943 (0.716)	-2.085 (0.723)
		3	-1.522 (0.727)	1.892 (0.752)	0.280 (0.740)	3.647 (0.765)	-1.532 (0.727)
		75%	0.25	-2.097 (0.730)	-5.076 (0.708)	-6.501 (0.698)	-6.486 (0.698)
		0.7	-2.247 (0.730)	-0.633 (0.742)	-4.009 (0.717)	-3.767 (0.719)	-2.154 (0.731)
		1.6	-1.288 (0.736)	1.287 (0.755)	-1.432 (0.735)	0.028 (0.746)	-1.282 (0.736)
		3	-1.971 (0.731)	-3.069 (0.723)	-2.235 (0.729)	1.463 (0.757)	-1.944 (0.731)
		0.14	0.25	-0.649 (0.733)	-13.656 (0.637)	-17.117 (0.612)	-17.145 (0.611)
		0.7	-1.957 (0.723)	0.554 (0.742)	-9.488 (0.668)	-9.534 (0.668)	-1.827 (0.724)
0.14	25%	1.6	-0.599 (0.733)	14.845 (0.847)	-0.774 (0.732)	0.893 (0.744)	-0.610 (0.733)
		3	-1.611 (0.726)	-1.286 (0.728)	-1.491 (0.727)	5.275 (0.777)	-2.028 (0.723)
		75%	0.25	-2.935 (0.724)	-4.639 (0.711)	-5.494 (0.705)	-5.474 (0.705)
		0.7	-0.303 (0.743)	0.786 (0.751)	-1.402 (0.735)	-1.184 (0.737)	-0.211 (0.744)
		1.6	-0.441 (0.742)	1.213 (0.755)	-0.505 (0.742)	0.558 (0.750)	-0.366 (0.743)
		3	-2.686 (0.725)	-3.438 (0.719)	-2.945 (0.723)	-0.144 (0.744)	-2.684 (0.725)

**Table E.3:** Coverage with MCSE for the log hazard ratio for sample size 100 and interval width 12 months

$\beta$	$S_0(60)$	$\gamma$	Coverage (MCSE)				
			Exact	Beg	Mid	End	IC
0.92	25%	0.25	94.67 (0.23)	76.85 (0.44)	71.00 (0.47)	71.39 (0.47)	94.92 (0.23)
		0.7	94.68 (0.23)	93.79 (0.25)	85.03 (0.37)	84.07 (0.38)	94.64 (0.23)
		1.6	94.59 (0.24)	89.29 (0.32)	94.57 (0.24)	94.43 (0.24)	94.64 (0.23)
		3	94.62 (0.24)	94.29 (0.24)	94.87 (0.23)	94.46 (0.24)	94.57 (0.24)
		75%	95.15 (0.22)	94.36 (0.24)	93.84 (0.25)	93.85 (0.25)	95.12 (0.22)
		0.25	95.20 (0.22)	95.63 (0.21)	94.74 (0.23)	94.77 (0.23)	95.21 (0.22)
		0.7	95.34 (0.22)	95.96 (0.21)	95.27 (0.22)	95.68 (0.21)	95.34 (0.22)
		1.6	95.33 (0.22)	94.96 (0.23)	95.27 (0.22)	95.84 (0.21)	95.33 (0.22)
		3	95.08 (0.23)	91.25 (0.29)	89.83 (0.32)	89.76 (0.32)	95.10 (0.23)
		75%	94.40 (0.24)	95.02 (0.23)	92.22 (0.28)	92.18 (0.28)	94.46 (0.24)
0.14	25%	0.25	94.93 (0.23)	97.66 (0.16)	94.85 (0.23)	95.14 (0.22)	94.91 (0.23)
		0.7	94.52 (0.24)	94.76 (0.23)	94.65 (0.23)	95.95 (0.21)	94.53 (0.24)
		1.6	95.46 (0.22)	94.91 (0.23)	94.72 (0.23)	94.77 (0.23)	95.39 (0.22)
		3	96.08 (0.20)	96.15 (0.20)	95.60 (0.21)	95.66 (0.21)	95.95 (0.21)
		75%	95.91 (0.21)	96.38 (0.19)	95.88 (0.21)	96.13 (0.20)	95.95 (0.21)
		0.25	95.10 (0.23)	94.85 (0.23)	94.97 (0.23)	95.74 (0.21)	95.05 (0.23)
		0.7	95.10 (0.23)	94.85 (0.23)	94.97 (0.23)	95.74 (0.21)	95.05 (0.23)
		1.6	95.10 (0.23)	94.85 (0.23)	94.97 (0.23)	95.74 (0.21)	95.05 (0.23)
		3	95.10 (0.23)	94.85 (0.23)	94.97 (0.23)	95.74 (0.21)	95.05 (0.23)
		75%	95.10 (0.23)	94.85 (0.23)	94.97 (0.23)	95.74 (0.21)	95.05 (0.23)

### E.3.2 Survival Probability at 12 and 48 Months

**Table E.4:** Bias with MCSE for the survival probability at 12 and 48 months (in the control group) for  $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100

Time point	$S_0(60)$	Width	Bias (MCSE)				
			Exact	Beg	Mid	End	IC
12 months	25%	3	-0.0001 (0.0006)	-0.0210 (0.0007)	0.1246 (0.0008)	0.1935 (0.0007)	0.0007 (0.0007)
		6	0.0001 (0.0006)	-0.0391 (0.0008)	0.1831 (0.0007)	0.2856 (0.0006)	0.0010 (0.0007)
		12	-0.0000 (0.0006)	-0.0630 (0.0008)	0.2734 (0.0007)	0.4179 (0.0005)	0.0014 (0.0007)
	75%	3	0.0001 (0.0005)	-0.0046 (0.0005)	0.0407 (0.0004)	0.0575 (0.0003)	-0.0006 (0.0005)
		6	0.0001 (0.0005)	-0.0104 (0.0005)	0.0537 (0.0004)	0.0776 (0.0003)	-0.0007 (0.0005)
		12	0.0001 (0.0005)	-0.0170 (0.0005)	0.0727 (0.0003)	0.1073 (0.0002)	-0.0010 (0.0005)
48 months	25%	3	-0.0002 (0.0006)	-0.0424 (0.0007)	-0.0566 (0.0007)	-0.0533 (0.0007)	-0.0004 (0.0006)
		6	-0.0001 (0.0006)	-0.0550 (0.0007)	-0.0588 (0.0007)	-0.0469 (0.0008)	-0.0002 (0.0006)
		12	-0.0002 (0.0006)	-0.0740 (0.0007)	-0.0563 (0.0008)	-0.0245 (0.0008)	-0.0002 (0.0006)
	75%	3	-0.0010 (0.0006)	-0.0040 (0.0006)	0.0023 (0.0006)	0.0059 (0.0006)	-0.0009 (0.0006)
		6	-0.0010 (0.0006)	-0.0058 (0.0006)	0.0047 (0.0006)	0.0109 (0.0006)	-0.0009 (0.0006)
		12	-0.0010 (0.0006)	-0.0083 (0.0006)	0.0089 (0.0006)	0.0207 (0.0006)	-0.0009 (0.0006)

**Table E.5:** Relative percentage error of the model-based standard errors with MCSE for the survival probability at 12 and 48 months (in the control group) for  $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100

Time point	$S_0(60)$	Width	Relative % error in ModSE (MCSE)				
			Exact	Beg	Mid	End	IC
12 months	25%	3	-2.808 (0.722)	-15.188 (0.630)	-20.882 (0.587)	-21.089 (0.586)	-2.917 (0.721)
		6	-2.862 (0.721)	-18.026 (0.610)	-22.405 (0.576)	-21.195 (0.587)	-2.763 (0.722)
		12	-2.969 (0.720)	-22.348 (0.579)	-23.211 (0.572)	-18.979 (0.609)	-3.133 (0.719)
	75%	3	-1.474 (0.738)	-1.835 (0.735)	0.368 (0.755)	2.330 (0.772)	-0.937 (0.743)
		6	-1.367 (0.739)	-2.260 (0.731)	1.786 (0.767)	5.703 (0.800)	-0.959 (0.744)
		12	-1.496 (0.738)	-3.112 (0.725)	4.646 (0.791)	13.699 (0.867)	-1.009 (0.744)
48 months	25%	3	-2.209 (0.731)	-13.868 (0.649)	-21.883 (0.592)	-23.456 (0.580)	-2.158 (0.731)
		6	-2.264 (0.730)	-16.655 (0.630)	-24.339 (0.574)	-25.511 (0.564)	-1.969 (0.732)
		12	-2.379 (0.730)	-20.997 (0.601)	-27.100 (0.554)	-27.497 (0.547)	-2.242 (0.730)
	75%	3	-1.552 (0.733)	-2.699 (0.725)	-3.971 (0.716)	-4.174 (0.715)	-1.439 (0.734)
		6	-1.447 (0.734)	-2.978 (0.723)	-4.234 (0.714)	-4.334 (0.714)	-1.329 (0.735)
		12	-1.596 (0.733)	-3.749 (0.717)	-4.755 (0.711)	-4.582 (0.712)	-1.454 (0.734)

**Table E.6:** Coverage with MCSE for the survival probability at 12 and 48 months (in the control group) for  $\gamma = 0.25$ , log hazard ratio 0.92 and sample size 100

Time point	$S_0(60)$	Width	Coverage (MCSE)				
			Exact	Beg	Mid	End	IC
12 months	25%	3	94.55 (0.24)	88.31 (0.34)	45.34 (0.52)	12.76 (0.35)	94.41 (0.24)
		6	94.53 (0.24)	83.09 (0.39)	16.64 (0.39)	0.54 (0.08)	94.65 (0.24)
		12	94.47 (0.24)	74.25 (0.46)	1.03 (0.11)	0.00 (0.00)	94.50 (0.24)
	75%	3	95.32 (0.22)	94.77 (0.23)	87.47 (0.35)	73.75 (0.46)	95.43 (0.22)
		6	95.38 (0.22)	93.66 (0.25)	77.46 (0.44)	46.28 (0.52)	95.26 (0.22)
		12	95.46 (0.22)	92.07 (0.28)	53.82 (0.52)	6.05 (0.25)	95.21 (0.22)
48 months	25%	3	94.38 (0.24)	82.37 (0.40)	74.01 (0.46)	74.64 (0.46)	94.59 (0.24)
		6	94.47 (0.24)	76.34 (0.44)	72.36 (0.47)	76.24 (0.45)	94.57 (0.24)
		12	94.40 (0.24)	66.45 (0.49)	71.83 (0.47)	81.05 (0.41)	94.64 (0.24)
	75%	3	95.30 (0.22)	94.82 (0.23)	94.95 (0.23)	94.97 (0.23)	95.25 (0.22)
		6	95.42 (0.22)	94.62 (0.24)	95.07 (0.23)	95.00 (0.23)	95.38 (0.22)
		12	95.36 (0.22)	94.03 (0.25)	94.99 (0.23)	94.61 (0.24)	95.28 (0.22)

# Appendix F

---

## Additional Material for Chapter 4

---

This appendix provides additional tables for the simulation study performed in Chapter 4: Stabilised versus unstabilised weights in an inverse probability weighted survival analysis.

## F.1 Simulation Study: Additional Tables

**Table F.1:** Bias (MCSE) of the marginal log hazard ratio  $\beta$ . Note that  $n_{sim} = 2500$  for  $n_{obs} = 2000$  and  $n_{sim} = 1000$  for  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	$n_{obs}$	Weibull		Cox	
				Unstabilised	Stabilised	Unstabilised	Stabilised
0.1	0.5	-	2000	-0.0022 (0.0029)	-0.0023 (0.0029)	-0.0041 (0.0029)	-0.0022 (0.0029)
			10000	-0.0025 (0.0020)	-0.0024 (0.0020)	-0.0028 (0.0020)	-0.0024 (0.0020)
			2000	0.0015 (0.0034)	0.0013 (0.0034)	-0.0002 (0.0034)	0.0015 (0.0034)
	0.05	-	10000	-0.0012 (0.0023)	-0.0013 (0.0024)	-0.0015 (0.0023)	-0.0013 (0.0023)
			2000	0.0186 (0.0028)	0.0135 (0.0026)	0.0113 (0.0028)	0.0124 (0.0026)
			10000	0.0039 (0.0020)	0.0031 (0.0019)	0.0025 (0.0021)	0.0029 (0.0019)
	0.05	-	2000	0.0069 (0.0030)	0.0023 (0.0029)	0.0003 (0.0029)	0.0011 (0.0029)
			10000	0.0047 (0.0022)	0.0035 (0.0021)	0.0039 (0.0022)	0.0035 (0.0021)
			2000	-0.0021 (0.0011)	-0.0021 (0.0011)	-0.0015 (0.0011)	-0.0015 (0.0011)
0.5	0.5	-	10000	0.0017 (0.0008)	0.0017 (0.0008)	0.0018 (0.0008)	0.0018 (0.0008)
			2000	-0.0017 (0.0013)	-0.0017 (0.0013)	-0.0010 (0.0013)	-0.0010 (0.0013)
			10000	0.0010 (0.0009)	0.0010 (0.0009)	0.0012 (0.0010)	0.0012 (0.0010)
	0.05	-	2000	0.0026 (0.0011)	0.0026 (0.0011)	0.0024 (0.0011)	0.0024 (0.0011)
			10000	0.0009 (0.0008)	0.0009 (0.0008)	0.0006 (0.0008)	0.0006 (0.0008)
			2000	0.0029 (0.0013)	0.0029 (0.0013)	0.0026 (0.0013)	0.0026 (0.0013)
	0.05	-	10000	0.0006 (0.0009)	0.0006 (0.0009)	0.0005 (0.0009)	0.0005 (0.0009)
			2000				

**Table F.2:** Bias (MCSE) of the difference in marginal RMST  $\Delta_\mu(20)$ . Note that  $n_{sim} = 2500$  for  $n_{obs} = 2000$  and  $n_{sim} = 1000$  for  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	$n_{obs}$	Weibull		Kaplan-Meier	
				Unstabilised	Stabilised	Unstabilised	Stabilised
0.1	0.5	-	2000	-0.0170 (0.0137)	-0.0169 (0.0137)	-0.0044 (0.0144)	-0.0044 (0.0144)
			10000	0.0055 (0.0094)	0.0055 (0.0094)	0.0072 (0.0100)	0.0072 (0.0100)
		0.05	2000	-0.0479 (0.0162)	-0.0466 (0.0163)	-0.0446 (0.0167)	-0.0446 (0.0167)
	2	-	2000	-0.0018 (0.0111)	-0.0012 (0.0111)	-0.0057 (0.0111)	-0.0057 (0.0111)
			10000	-0.0776 (0.0162)	-0.0568 (0.0157)	-0.0420 (0.0156)	-0.0420 (0.0156)
		0.05	2000	-0.0170 (0.0119)	-0.0139 (0.0114)	-0.0127 (0.0111)	-0.0127 (0.0111)
0.5	0.5	-	2000	-0.0043 (0.0173)	0.0115 (0.0170)	0.0271 (0.0179)	0.0271 (0.0179)
			10000	-0.0194 (0.0125)	-0.0154 (0.0122)	-0.0174 (0.0126)	-0.0174 (0.0126)
		0.05	2000	0.0090 (0.0060)	0.0090 (0.0060)	0.0116 (0.0069)	0.0116 (0.0069)
	2	-	2000	-0.0089 (0.0041)	-0.0089 (0.0041)	-0.0126 (0.0047)	-0.0126 (0.0047)
			10000	0.0057 (0.0072)	0.0057 (0.0072)	0.0068 (0.0075)	0.0068 (0.0075)
		0.05	2000	-0.0064 (0.0051)	-0.0064 (0.0051)	-0.0051 (0.0053)	-0.0051 (0.0053)
0.5	0.5	-	2000	-0.0094 (0.0066)	-0.0094 (0.0066)	-0.0112 (0.0069)	-0.0112 (0.0069)
			10000	-0.0043 (0.0046)	-0.0043 (0.0046)	-0.0047 (0.0049)	-0.0047 (0.0049)
		0.05	2000	-0.0118 (0.0076)	-0.0118 (0.0076)	-0.0159 (0.0078)	-0.0159 (0.0078)
	2	-	2000	-0.0023 (0.0051)	-0.0023 (0.0051)	-0.0018 (0.0053)	-0.0018 (0.0053)
			10000				

# Appendix G

---

## Additional Material for Chapter 5

---

This appendix provides additional material for Chapter 5: Closed-form variance estimator for inverse probability weighted parametric survival models. The appendix begins by proposing the M-estimation variance estimator for an IP weighted exponential, Weibull, Gompertz, log-logistic and log-normal survival model. This corresponds to the methods described in Section 5.4. Appendix G.2 gives the starting seed for the simulation study, as referenced in Section 5.5.6. Appendix G.3 and G.4 give additional figures and tables for the simulation study, respectively. This includes both the large and small sample results.

## G.1 M-estimation Variance Estimator for IP Weighted Parametric Models

### G.1.1 Exponential Model

Recall the formulas for an exponential model from Section 2.3.3. In particular, the log-likelihood was given in Equation 2.10. This is repeated below but assuming one covariate (treatment) with the corresponding coefficient  $\beta_1$ , where  $\beta_1$  is the log hazard ratio. Note that  $\beta_0$  in Equation 2.10 is now represented as  $\beta_2$ .

$$l_i(\beta_1, \beta_2 | t_i, \delta_i, z_i) = \delta_i \log(\lambda_i) - \lambda_i t_i$$

$$\lambda_i = \exp(\beta_2 + \beta_1 z_i)$$

Following the notation in Section 5.4,  $\boldsymbol{\beta} = (\beta_1, \beta_2)$ . Following Equation 2.25, the weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,  $s_i$ , as defined in Equation 2.24:

$$l_i^w(\boldsymbol{\beta} | t_i, \delta_i, z_i, w_i) = w_i \{ \delta_i \log(\lambda_i) - \lambda_i t_i \} \quad w_i = s_i, u_i$$

The estimating equations from Equation 5.1 for unstabilised weights become the following. The estimating equations from Equation 5.2 can be written similarly for stabilised weights.

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{ Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha}) \} \\ u_i Z_i (\delta_i - \lambda_i T_i) \\ u_i (\delta_i - \lambda_i T_i) \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights

is estimated as follows (similar for stabilised weights):

$$\mathbf{A}_3 = \widehat{u}_i \begin{pmatrix} z_i^2 \widehat{\lambda}_i t_i \\ z_i \widehat{\lambda}_i t_i & \widehat{\lambda}_i t_i \end{pmatrix}$$

### G.1.2 Weibull Model

Recall the formulas for a Weibull model from Section 2.3.3. In particular, the log-likelihood was given in Equation 2.12. This is repeated below but assuming one covariate (treatment) with the corresponding coefficient  $\beta_1$ , where  $\beta_1$  is the log hazard ratio. Note that  $\beta_0$  in Equation 2.12 is now represented as  $\beta_2$ .

$$l_i(\beta_1, \beta_2, \gamma | t_i, \delta_i, z_i) = \delta_i \log(\gamma \lambda_i t_i^{\gamma-1}) - \lambda_i t_i^\gamma$$

$$\lambda_i = \exp(\beta_2 + \beta_1 z_i)$$

Following Equation 2.25, the weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,  $s_i$ , as defined in Equation 2.24:

$$l_i^w(\beta_1, \beta_2, \gamma | t_i, \delta_i, z_i, w_i) = w_i \{ \delta_i \log(\gamma \lambda_i t_i^{\gamma-1}) - \lambda_i t_i^\gamma \} \quad w_i = s_i, u_i$$

$\gamma$  is modelled on the log scale to ensure positivity. Following the notation in Section 5.4, this makes  $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3 = \log(\gamma))$ . Reparameterising the equation above gives the following weighted log-likelihood contribution for individual  $i$ :

$$l_i^w(\boldsymbol{\beta} | t_i, \delta_i, z_i, w_i) = w_i \left( \delta_i [\beta_3 + \log(\lambda_i) + \{\exp(\beta_3) - 1\} \log(t_i)] - \lambda_i t_i^{\exp(\beta_3)} \right)$$

$$w_i = s_i, u_i$$

The estimating equations from Equation 5.1 for unstabilised weights become the following. The estimating equations from Equation 5.2 can be written similarly for

stabilised weights.

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ u_i Z_i \left\{ \delta_i - \lambda_i T_i^{\exp(\beta_3)} \right\} \\ u_i \left\{ \delta_i - \lambda_i T_i^{\exp(\beta_3)} \right\} \\ u_i \left[ \delta_i \{1 + \exp(\beta_3) \log(T_i)\} - \lambda_i T_i^{\exp(\beta_3)} \exp(\beta_3) \log(T_i) \right] \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights is estimated as follows (similar for stabilised weights):

$$\mathbf{A}_3 = \widehat{u}_i \begin{pmatrix} z_i^2 \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} & & & \\ z_i \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} & \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} & & \\ z_i \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} \exp(\widehat{\beta}_3) \log(t_i) & \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} \exp(\widehat{\beta}_3) \log(t_i) & -\frac{\partial^2 l_i^u}{\partial \beta_3^2} \Big|_{\widehat{\boldsymbol{\theta}}} & \\ & & & \end{pmatrix} \quad (\text{G.1})$$

$$-\frac{\partial^2 l_i^u}{\partial \beta_3^2} \Big|_{\widehat{\boldsymbol{\theta}}} = -\delta_i \exp(\widehat{\beta}_3) \log(t_i) + \widehat{\lambda}_i t_i^{\exp(\widehat{\beta}_3)} \exp(\widehat{\beta}_3) \log(t_i) \left\{ \exp(\widehat{\beta}_3) \log(t_i) + 1 \right\}$$

### G.1.3 Gompertz Model

Recall the formulas for a Gompertz model from Section 2.3.3. In particular, the log-likelihood was given in Equation 2.13. This is repeated below but assuming one covariate (treatment) with the corresponding coefficient  $\beta_1$ , where  $\beta_1$  is the log hazard ratio. Note that  $\beta_0$  in Equation 2.13 is now represented as  $\beta_2$ .

$$l_i(\beta_1, \beta_2, \gamma | t_i, \delta_i, z_i) = \delta_i \{\log(\lambda_i) + \gamma t_i\} - \lambda_i \gamma^{-1} \{\exp(\gamma t_i) - 1\}$$

$$\lambda_i = \exp(\beta_2 + \beta_1 z_i)$$

Following the notation in Section 5.4,  $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3 = \gamma)$ . Following Equation 2.25, the weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,

$s_i$ , as defined in Equation 2.24:

$$l_i^w(\boldsymbol{\beta}|t_i, \delta_i, z_i, w_i) = w_i [\delta_i \{\log(\lambda_i) + \beta_3 t_i\} - \lambda_i \beta_3^{-1} \{\exp(\beta_3 t_i) - 1\}] \quad w_i = s_i, u_i$$

The estimating equations from Equation 5.1 for unstabilised weights become the following. The estimating equations from Equation 5.2 can be written similarly for stabilised weights.

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ u_i Z_i [\delta_i - \lambda_i \beta_3^{-1} \{\exp(\beta_3 T_i) - 1\}] \\ u_i [\delta_i - \lambda_i \beta_3^{-1} \{\exp(\beta_3 T_i) - 1\}] \\ u_i [\delta_i T_i - \lambda_i \beta_3^{-2} \{\exp(\beta_3 T_i) (\beta_3 T_i - 1) + 1\}] \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights is estimated as follows (similar for stabilised weights). For ease of displaying the formulas, let  $\hat{v}_i = \exp(\hat{\beta}_3 t_i)$ .

$$\mathbf{A}_3 = \hat{u}_i \begin{pmatrix} z_i^2 \hat{\lambda}_i \hat{\beta}_3^{-1} (\hat{v}_i - 1) & & \\ z_i \hat{\lambda}_i \hat{\beta}_3^{-1} (\hat{v}_i - 1) & \hat{\lambda}_i \hat{\beta}_3^{-1} (\hat{v}_i - 1) & \\ z_i \hat{\lambda}_i \hat{\beta}_3^{-2} \left\{ \hat{v}_i (\hat{\beta}_3 t_i - 1) + 1 \right\} & \hat{\lambda}_i \hat{\beta}_3^{-2} \left\{ \hat{v}_i (\hat{\beta}_3 t_i - 1) + 1 \right\} & -\frac{\partial^2 l_i^u}{\partial \beta_3^2} \Big|_{\hat{\boldsymbol{\theta}}} \end{pmatrix}$$

$$-\frac{\partial^2 l_i^u}{\partial \beta_3^2} \Big|_{\hat{\boldsymbol{\theta}}} = \hat{\lambda}_i \hat{\beta}_3^{-3} \left\{ \hat{v}_i \left( \hat{\beta}_3^2 t_i^2 - 2\hat{\beta}_3 t_i + 2 \right) - 2 \right\}$$

#### G.1.4 Log-logistic Model

Recall the formulas for a log-logistic model from Section 2.3.5. In particular, the log-likelihood was given in Equation 2.15. This is repeated below but assuming one covariate (treatment) with the corresponding coefficient  $\beta_1$ , where  $\beta_1$  is the log time

ratio. Note that  $\alpha_0$  in Equation 2.15 is now represented as  $\beta_2$ .

$$l_i(\beta_1, \beta_2, \gamma | t_i, \delta_i, z_i) = \delta_i \left\{ \frac{1}{\gamma} \log(\lambda_i) + \left( \frac{1}{\gamma} - 1 \right) \log(t_i) - \log(\gamma) \right\} - \\ (\delta_i + 1) \log \left\{ 1 + (\lambda_i t_i)^{1/\gamma} \right\} \\ \lambda_i = \exp(-\beta_2 - \beta_1 z_i)$$

The weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,  $s_i$ , as defined in Equation 2.24:

$$l_i^w(\beta_1, \beta_2, \gamma | t_i, \delta_i, z_i, w_i) = w_i \delta_i \left\{ \frac{1}{\gamma} \log(\lambda_i) + \left( \frac{1}{\gamma} - 1 \right) \log(t_i) - \log(\gamma) \right\} - \\ w_i (\delta_i + 1) \log \left\{ 1 + (\lambda_i t_i)^{1/\gamma} \right\} \quad w_i = s_i, u_i$$

$\gamma$  is modelled on the log scale to ensure positivity. Following the notation in Section 5.4, this makes  $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3 = \log(\gamma))$ . Reparameterising the equation above gives the following weighted log-likelihood contribution for individual  $i$ :

$$l_i^w(\boldsymbol{\beta} | t_i, \delta_i, z_i, w_i) = w_i \delta_i [\exp(-\beta_3) \log(\lambda_i) + \{\exp(-\beta_3) - 1\} \log(t_i) - \beta_3] - \\ w_i (\delta_i + 1) \log \left\{ 1 + (\lambda_i t_i)^{\exp(-\beta_3)} \right\} \quad w_i = s_i, u_i$$

The estimating equations from Equation 5.1 for unstabilised weights become the following. The estimating equations from Equation 5.2 can be written similarly for stabilised weights. For ease of displaying the formulas, let  $v_i = (\lambda_i T_i)^{\exp(-\beta_3)}$ .

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ u_i Z_i \exp(-\beta_3) \left( -\delta_i + \frac{\delta_i + 1}{1 + v_i^{-1}} \right) \\ u_i \exp(-\beta_3) \left( -\delta_i + \frac{\delta_i + 1}{1 + v_i^{-1}} \right) \\ u_i \left\{ \exp(-\beta_3) \log(\lambda_i T_i) \left( -\delta_i + \frac{\delta_i + 1}{1 + v_i^{-1}} \right) - \delta_i \right\} \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights

is estimated as follows (similar for stabilised weights), where  $v_i$  now depends on observed value  $t_i$ :

$$\begin{aligned}\widehat{q}_i &= \frac{\exp(-2\widehat{\beta}_3)(\delta_i + 1)}{\widehat{v}_i(1 + \widehat{v}_i^{-1})^2} \\ \widehat{r}_i &= \exp(-\widehat{\beta}_3) \left[ -\delta_i + \frac{(\delta_i + 1) \left\{ \exp(-\widehat{\beta}_3) \widehat{v}_i^{-1} \log(\widehat{\lambda}_i t_i) + 1 + \widehat{v}_i^{-1} \right\}}{(1 + \widehat{v}_i^{-1})^2} \right] \\ \mathbf{A}_3 &= \widehat{u}_i \begin{pmatrix} z_i^2 \widehat{q}_i \\ z_i \widehat{q}_i & \widehat{q}_i \\ z_i \widehat{r}_i & \widehat{r}_i & \log(\widehat{\lambda}_i t_i) \widehat{r}_i \end{pmatrix}\end{aligned}$$

### G.1.5 Log-normal Model

Recall the formulas for a log-logistic model from Section 2.3.5. In particular, the log-likelihood was given in Equation 2.16. This is repeated below but assuming one covariate (treatment) with the corresponding coefficient  $\beta_1$ , where  $\beta_1$  is the log time ratio. Note that  $\alpha_0$  in Equation 2.16 is now represented as  $\beta_2$ .

$$\begin{aligned}l_i(\beta_1, \beta_2, \sigma | t_i, \delta_i, z_i) &= \delta_i \left[ -\log(t_i) - \log(\sigma) - \frac{1}{2} \log(2\pi) - \frac{1}{2\sigma^2} \{\log(t_i) - \mu_i\}^2 \right] + \\ &\quad (1 - \delta_i) \log \left[ 1 - \Phi \left\{ \frac{\log(t_i) - \mu_i}{\sigma} \right\} \right] \\ \mu_i &= \beta_2 + \beta_1 z_i\end{aligned}$$

The weighted log-likelihood contribution for individual  $i$  is as follows, where weight  $w_i$  is either unstabilised,  $u_i$ , as defined in Equation 2.23 or stabilised,  $s_i$ , as defined in Equation 2.24:

$$\begin{aligned}l_i^w(\beta_1, \beta_2, \sigma | t_i, \delta_i, z_i, w_i) &= w_i \delta_i \left[ -\log(t_i) - \log(\sigma) - \frac{1}{2} \log(2\pi) - \frac{1}{2\sigma^2} \{\log(t_i) - \mu_i\}^2 \right] \\ &\quad + w_i (1 - \delta_i) \log \left[ 1 - \Phi \left\{ \frac{\log(t_i) - \mu_i}{\sigma} \right\} \right] \quad w_i = s_i, u_i\end{aligned}$$

$\sigma$  is modelled on the log scale to ensure positivity. Following the notation in Section 5.4, this makes  $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3 = \log(\sigma))$ . Reparameterising the equation

above gives the following weighted log-likelihood contribution for individual  $i$ :

$$l_i^w(\boldsymbol{\beta}|t_i, \delta_i, z_i, w_i) = w_i \delta_i \left[ -\log(t_i) - \beta_3 - \frac{1}{2} \log(2\pi) - \frac{1}{2\exp(2\beta_3)} \{\log(t_i) - \mu_i\}^2 \right] + \\ w_i (1 - \delta_i) \log \left[ 1 - \Phi \left\{ \frac{\log(t_i) - \mu}{\exp(\beta_3)} \right\} \right] \quad w_i = s_i, u_i$$

The estimating equations from Equation 5.1 for unstabilised weights become the following, where  $\phi(\cdot)$  is the standard normal density function. The estimating equations from Equation 5.2 can be written similarly for stabilised weights. For ease of displaying the formulas, let  $v_i = \frac{\log(T_i) - \mu_i}{\exp(\beta_3)}$ .

$$\mathbf{u}(\boldsymbol{\theta}; T_i, \delta_i, Z_i, \mathbf{X}_i) = \begin{pmatrix} \mathbf{X}_i^T \{Z_i - e_i(\mathbf{X}_i, \boldsymbol{\alpha})\} \\ u_i Z_i \exp(-\beta_3) \left[ \delta_i \exp(-\beta_3) \{\log(T_i) - \mu_i\} + \frac{(1-\delta_i)\phi(v_i)}{1-\Phi(v_i)} \right] \\ u_i \exp(-\beta_3) \left[ \delta_i \exp(-\beta_3) \{\log(T_i) - \mu_i\} + \frac{(1-\delta_i)\phi(v_i)}{1-\Phi(v_i)} \right] \\ u_i \left( \delta_i [\exp(-2\beta_3) \{\log(T_i) - \mu_i\}^2 - 1] + \frac{(1-\delta_i)\phi(v_i)v_i}{1-\Phi(v_i)} \right) \end{pmatrix}$$

The M-estimation variance estimator is defined as given in Section 5.4.4 for unstabilised and stabilised weights. Symmetric matrix  $\mathbf{A}_3$  for unstabilised weights is estimated as follows (similar for stabilised weights), where  $v_i$  now depends on observed value  $t_i$ :

$$\widehat{m}_i = \exp(-2\widehat{\beta}_3) \left( \delta_i - \frac{(1-\delta_i) \left[ \{1 - \Phi(\widehat{v}_i)\} \frac{\widehat{v}_i^2}{\sqrt{2\pi}} \exp\left(-\frac{\widehat{v}_i^2}{2}\right) - \phi(\widehat{v}_i)^2 \right]}{\{1 - \Phi(\widehat{v}_i)\}^2} \right)$$

$$\widehat{q}_i = 2\delta_i \exp(-2\widehat{\beta}_3) \{\log(t_i) - \widehat{\mu}_i\}$$

$$\widehat{r}_i = \frac{(1-\delta_i) \left[ \{1 - \Phi(\widehat{v}_i)\} \left\{ \frac{\widehat{v}_i^2}{\sqrt{2\pi}} \exp\left(-\frac{\widehat{v}_i^2}{2}\right) - \phi(\widehat{v}_i) \right\} - \phi(\widehat{v}_i)^2 \widehat{v}_i \right]}{\{1 - \Phi(\widehat{v}_i)\}^2}$$

$$\mathbf{A}_3 = \widehat{u}_i \begin{pmatrix} z_i^2 \widehat{m}_i & & \\ & z_i \widehat{m}_i & \widehat{m}_i \\ z_i \left\{ \widehat{q}_i - \exp(-\widehat{\beta}_3) \widehat{r}_i \right\} & \widehat{q}_i - \exp(-\widehat{\beta}_3) \widehat{r}_i & \widehat{q}_i \{\log(t_i) - \widehat{\mu}_i\} - \widehat{v}_i \widehat{r}_i \end{pmatrix}$$

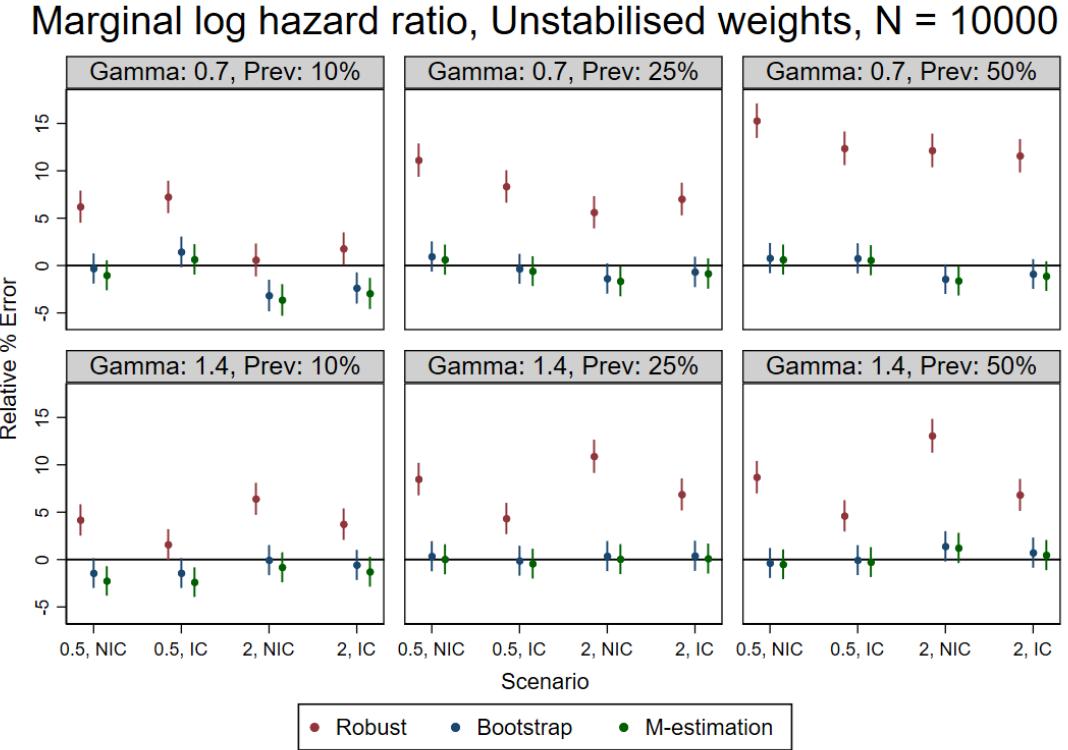
## G.2 Simulation Study: Starting Seeds

**Table G.1:** Starting seeds for the preliminary and main simulation studies in Chapter 5 for each of the 24 batches

Batch	$\gamma, \lambda$	$\pi_Z$	$e^\beta$	$\lambda_C$	Preliminary Seed	Main Seed
1	0.7, 0.15	0.1	0.5	-	68974653	8285889
2				0.05	92831467	442154
3			2	-	69967318	8630377
4				0.05	58266073	4526046
5		0.25	0.5	-	9822963	7720400
6				0.05	65929101	5861199
7			2	-	70364889	4227766
8				0.05	56090126	2729307
9		0.5	0.5	-	86491290	8053644
10				0.05	92904620	3060019
11			2	-	99863536	1190997
12				0.05	46156778	7247310
13	1.4, 0.003	0.1	0.5	-	64649368	6964866
14				0.05	99484752	9119345
15			2	-	34727798	6795634
16				0.05	60356018	3549416
17		0.25	0.5	-	34348778	7389700
18				0.05	35748674	1874017
19			2	-	78631317	3146128
20				0.05	14177532	1375693
21		0.5	0.5	-	28509040	6537739
22				0.05	13538605	2701319
23			2	-	56200428	8998394
24				0.05	15548638	5734232

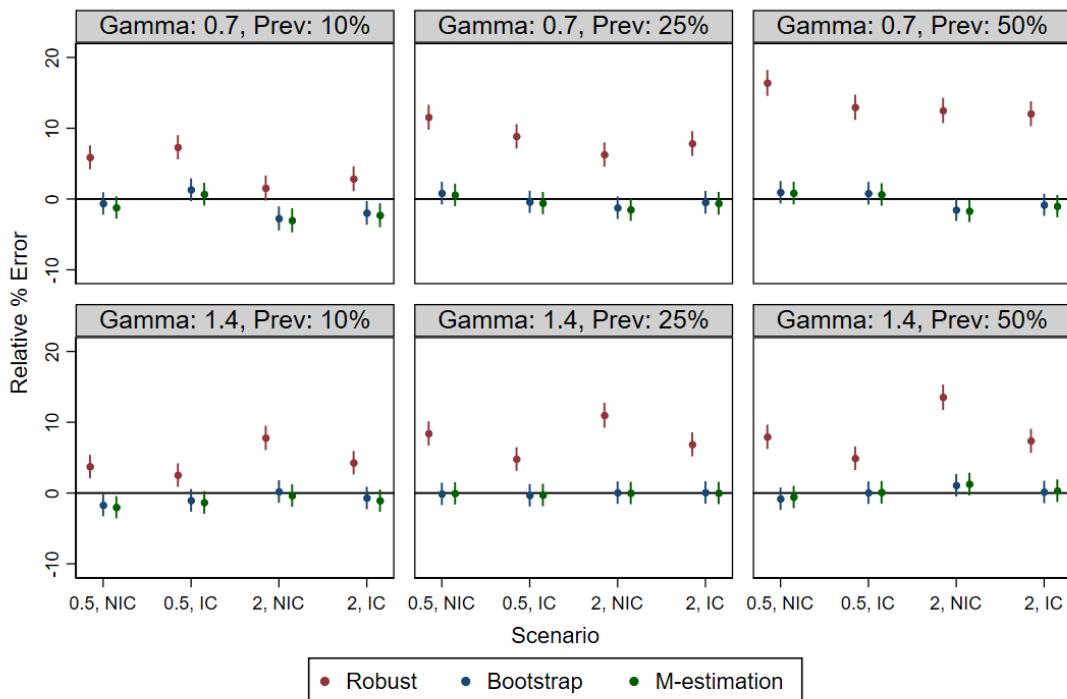
## G.3 Simulation Study: Additional Figures

### G.3.1 Large Samples



**Figure G.1:** The relative percentage error of the unstabilised variance estimators for the marginal log hazard ratio for the large sample size  $n_{obs} = 10000$ . The unstabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

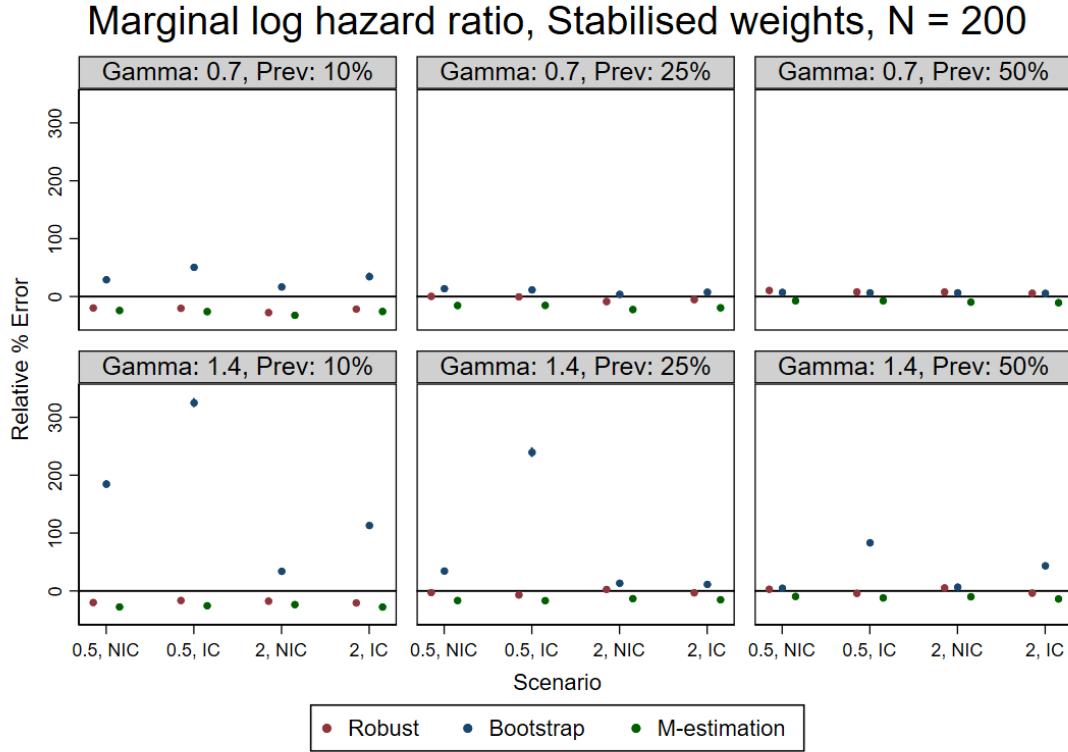
## Difference in marginal RMST, Unstabilised weights, N = 10000



**Figure G.2:** The relative percentage error of the unstabilised variance estimators for the difference in marginal RMST for the large sample size  $n_{obs} = 10000$ . The unstabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

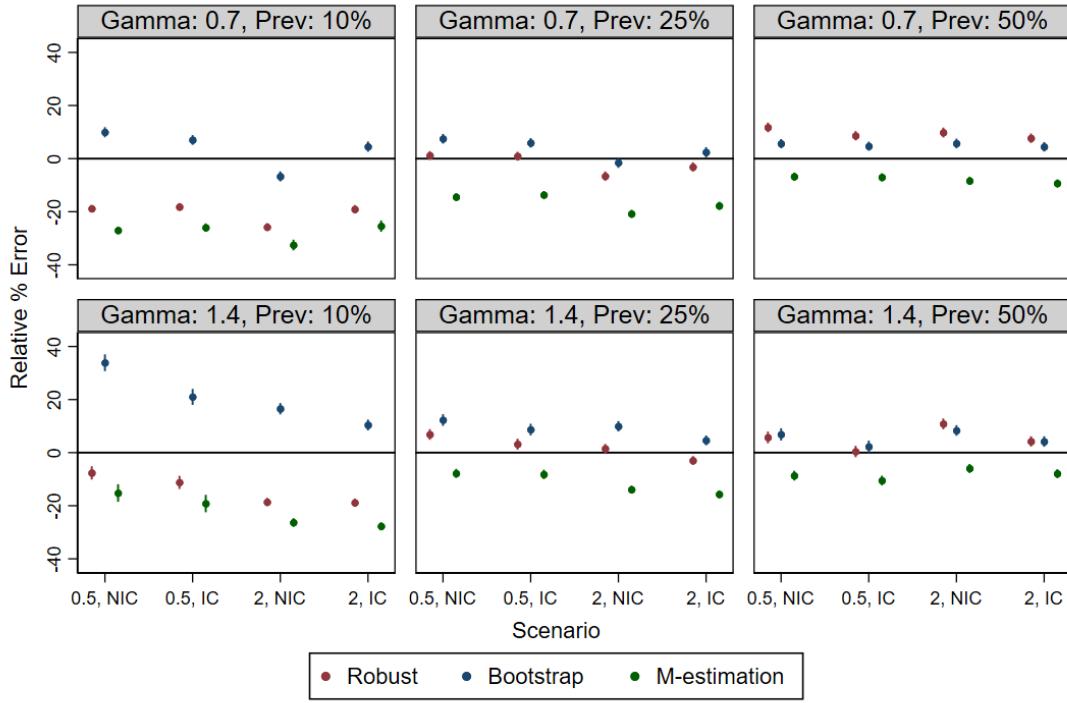
### G.3.2 Small Samples

The simulation study was designed to investigate the large sample properties of the proposed variance estimator. A sample size of 200 was also considered as an exploratory analysis. The exploratory results from the small sample simulation are shown here, excluding the estimates with the difficulties described in Section 5.6.3.



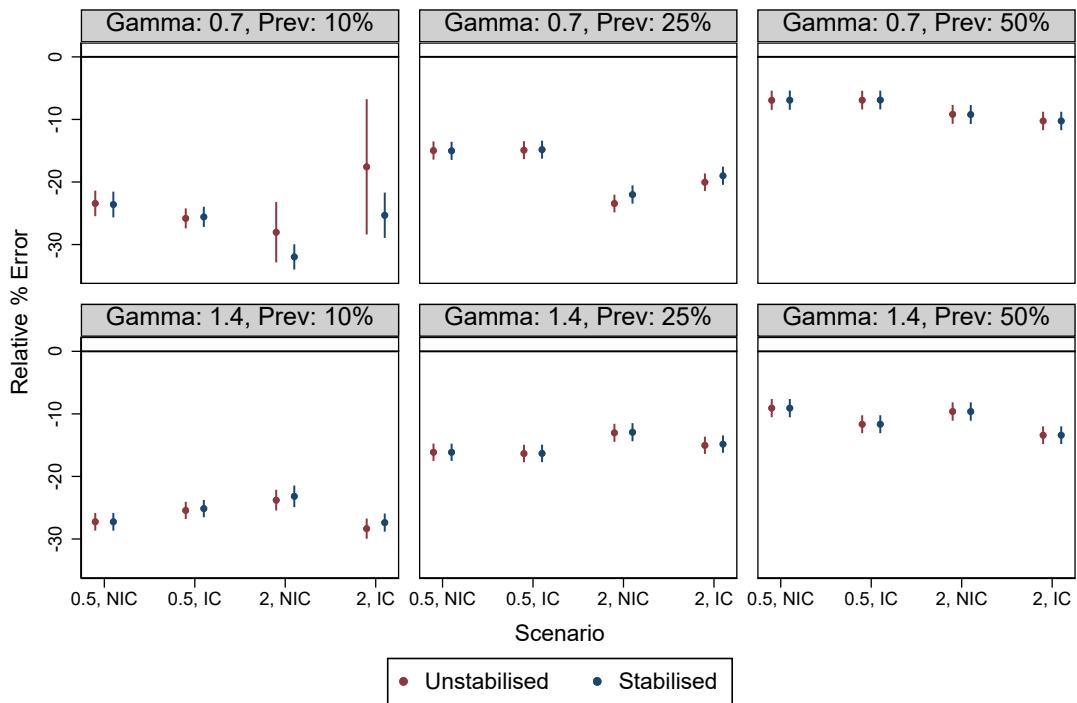
**Figure G.3:** The relative percentage error of the stabilised variance estimators for the marginal log hazard ratio for the small sample size  $n_{obs} = 200$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

## Difference in marginal RMST, Stabilised weights, N = 200



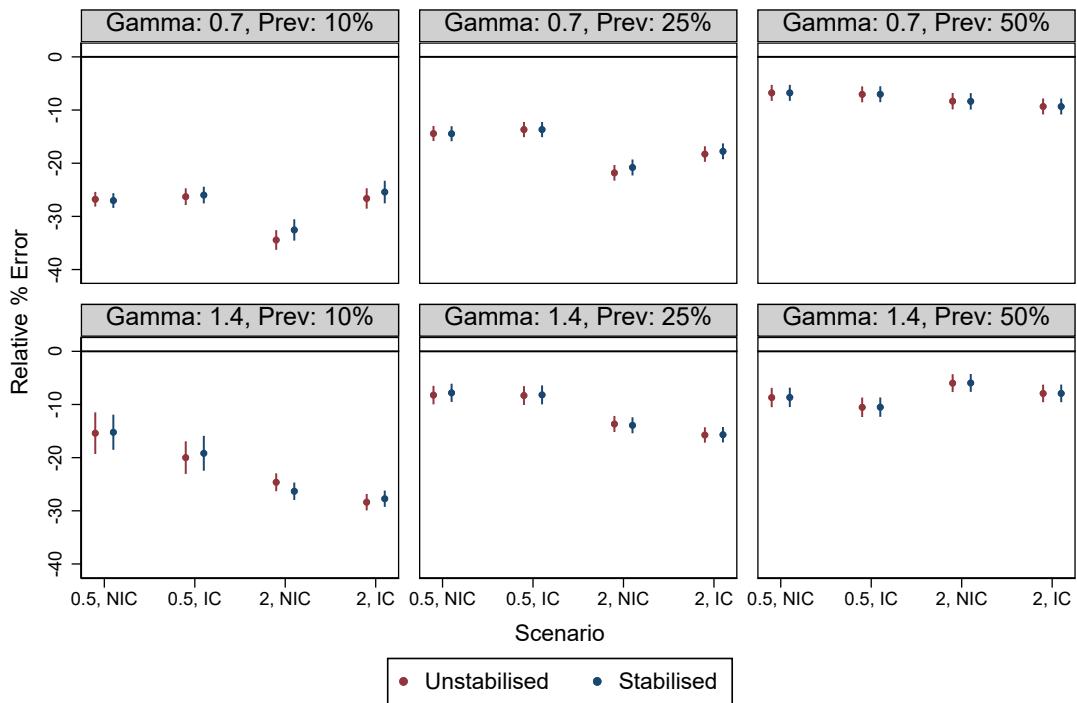
**Figure G.4:** The relative percentage error of the stabilised variance estimators for the difference in marginal RMST for the small sample size  $n_{obs} = 200$ . The stabilised robust, bootstrap and M-estimation variance estimators are shown in red, blue and green, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

### Marginal log hazard ratio, M-estimation, N = 200



**Figure G.5:** The relative percentage error of the M-estimation variance estimators for the marginal log hazard ratio for the small sample size  $n_{obs} = 200$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

## Difference in marginal RMST, M-estimation, N = 200



**Figure G.6:** The relative percentage error of the M-estimation variance estimators for the difference in marginal RMST for the small sample size  $n_{obs} = 200$ . The unstabilised and stabilised variance estimators are shown in red and blue, respectively. 95% confidence intervals are shown by vertical spikes. The top panels represent  $\gamma = 0.7$  and the bottom panels represent  $\gamma = 1.4$ . The first, second and third column show treatment prevalence  $\pi_Z = 0.1, 0.25$  and  $0.5$ , respectively. Within each graph, the first two scenarios represent a marginal hazard ratio  $\exp(\beta) = 0.5$  and the second two scenarios represent a marginal hazard ratio  $\exp(\beta) = 2$ . The first and third scenarios have no intermittent censoring (NIC), while the second and fourth scenarios have intermittent censoring (IC)

## G.4 Simulation Study: Additional Tables

### G.4.1 Large Samples

**Table G.2:** Relative % error of ModSE (MCSE) for the marginal log hazard ratio for  $\gamma = 0.7$ ,  $\lambda = 0.15$  and  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised			
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest
0.1	0.5	-	0.062	6.21 (0.87)	-0.31 (0.81)	-1.03 (0.81)	0.063	5.76 (0.86)	-0.35 (0.81)	-1.06 (0.81)
		0.05	0.072	7.24 (0.87)	1.44 (0.83)	0.65 (0.82)	0.073	6.95 (0.87)	1.45 (0.83)	0.68 (0.82)
		2	-	0.065 (0.89)	0.59 (0.85)	-3.16 (0.85)	-3.63	0.062 (0.90)	2.37 (0.86)	-2.46 (0.86)
	0.25	0.05	0.067	1.78 (0.88)	-2.37 (0.84)	-2.95 (0.84)	0.065	3.23 (0.89)	-1.87 (0.85)	-2.36 (0.85)
		0.5	-	0.035 (0.90)	11.12 (0.82)	0.95 (0.81)	0.63	0.035 (0.89)	10.67 (0.82)	0.94 (0.81)
		2	-	0.042 (0.88)	8.35 (0.81)	-0.34 (0.80)	-0.59	0.043 (0.87)	8.04 (0.81)	-0.34 (0.80)
0.5	0.5	-	0.036 (0.87)	5.62 (0.81)	-1.37 (0.81)	-1.66 (0.81)	0.035	6.71 (0.88)	-1.28 (0.81)	-1.57 (0.81)
		0.05	0.038	7.02 (0.88)	-0.67 (0.82)	-0.84 (0.82)	0.037	8.17 (0.89)	-0.36 (0.82)	-0.54 (0.82)
		2	-	0.024 (0.93)	15.29 (0.81)	0.79 (0.81)	0.63	0.024 (0.93)	15.29 (0.81)	0.79 (0.81)
	0.05	0.05	0.030	12.37 (0.91)	0.76 (0.81)	0.56 (0.81)	0.030	12.38 (0.91)	0.76 (0.81)	0.56 (0.81)
		2	-	0.025 (0.91)	12.15 (0.80)	-1.43 (0.80)	-1.60 (0.80)	0.025 (0.91)	12.15 (0.80)	-1.44 (0.80)
		0.05	0.028	11.59 (0.90)	-0.90 (0.80)	-1.11 (0.80)	0.028	11.59 (0.90)	-0.90 (0.80)	-1.12 (0.80)

**Table G.3:** Relative % error of ModSE (MCSE) for the marginal log hazard ratio for  $\gamma = 1.4$ ,  $\lambda = 0.003$  and  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised				
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest	
0.1	0.5	-	0.094	4.18 (0.84)	-1.43 (0.80)	-2.25 (0.79)	0.094	4.10 (0.84)	-1.46 (0.80)	-2.28 (0.79)	
		0.05	0.135	1.59 (0.83)	-1.43 (0.80)	-2.38 (0.80)	0.136	1.57 (0.83)	-1.43 (0.81)	-2.37 (0.80)	
	2	-	0.067	6.41 (0.86)	-0.06 (0.81)	-0.81 (0.81)	0.066	6.73 (0.87)	-0.04 (0.81)	-0.79 (0.81)	
		0.05	0.092	3.74 (0.85)	-0.57 (0.81)	-1.28 (0.81)	0.091	3.98 (0.85)	-0.55 (0.81)	-1.23 (0.81)	
	0.25	0.5	-	0.055	8.49 (0.88)	0.35 (0.81)	0.03 (0.81)	0.055	8.44 (0.88)	0.34 (0.81)	0.02 (0.81)
		0.05	0.082	4.33 (0.84)	-0.12 (0.81)	-0.43 (0.81)	0.082	4.31 (0.84)	-0.12 (0.81)	-0.43 (0.81)	
0.5	0.5	-	0.040	10.90 (0.90)	0.37 (0.81)	0.05 (0.81)	0.040	11.16 (0.90)	0.40 (0.81)	0.08 (0.81)	
		0.05	0.058	6.87 (0.86)	0.40 (0.81)	0.11 (0.81)	0.058	7.04 (0.86)	0.43 (0.81)	0.14 (0.81)	
	2	-	0.045	8.70 (0.88)	-0.36 (0.80)	-0.50 (0.80)	0.045	8.70 (0.88)	-0.36 (0.80)	-0.50 (0.80)	
		0.05	0.069	4.61 (0.84)	-0.06 (0.81)	-0.27 (0.80)	0.069	4.61 (0.84)	-0.06 (0.81)	-0.27 (0.80)	
	0.05	-	0.036	13.07 (0.91)	1.40 (0.82)	1.22 (0.82)	0.036	13.07 (0.91)	1.40 (0.82)	1.22 (0.82)	
		0.05	0.056	6.82 (0.86)	0.73 (0.81)	0.48 (0.81)	0.056	6.82 (0.86)	0.73 (0.81)	0.48 (0.81)	

**Table G.4:** Relative % error of ModSE (MCSE) for the difference in marginal RMST  $\Delta_\mu(20)$  for  $\gamma = 0.7$ ,  $\lambda = 0.15$  and  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised				
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest	
0.1	0.5	-	0.300	5.89 (0.86)	-0.63 (0.81)	-1.21 (0.80)	0.301 0.346	5.83 (0.86)	-0.66 (0.81)	-1.22 (0.80)	
		0.05	0.343	7.31 (0.87)	1.31 (0.82)	0.68 (0.82)	0.346 0.383	7.15 (0.87)	1.30 (0.82)	0.70 (0.82)	
	2	-	0.380	1.54 (0.91)	-2.74 (0.87)	-3.02 (0.87)	0.366 0.383	2.83 (0.92)	-2.19 (0.88)	-2.44 (0.89)	
		0.05	0.391	2.86 (0.90)	-1.96 (0.86)	-2.30 (0.86)	0.383 0.383	3.73 (0.91)	-1.58 (0.86)	-1.93 (0.87)	
	0.25	0.5	-	0.172	11.55 (0.90)	0.83 (0.82)	0.58 (0.81)	0.172 0.209	11.49 (0.90)	0.82 (0.82)	0.57 (0.81)
		0.05	0.208	8.86 (0.88)	-0.40 (0.81)	-0.58 (0.80)	0.209 0.221	8.69 (0.88)	-0.41 (0.81)	-0.57 (0.80)	
0.5	0.5	-	0.213	6.28 (0.88)	-1.23 (0.82)	-1.50 (0.81)	0.208 0.221	7.07 (0.88)	-1.16 (0.82)	-1.44 (0.81)	
		0.05	0.224	7.83 (0.89)	-0.46 (0.82)	-0.61 (0.82)	0.221 0.221	8.52 (0.89)	-0.27 (0.82)	-0.42 (0.82)	
	2	-	0.130	16.40 (0.94)	0.96 (0.81)	0.84 (0.81)	0.130 0.163	16.40 (0.94)	0.96 (0.81)	0.83 (0.81)	
		0.05	0.163	12.96 (0.91)	0.81 (0.81)	0.64 (0.81)	0.163 0.163	12.96 (0.91)	0.81 (0.81)	0.64 (0.81)	
	0.05	-	0.147	12.51 (0.91)	-1.54 (0.80)	-1.70 (0.79)	0.147 0.169	12.50 (0.91)	-1.55 (0.80)	-1.70 (0.79)	
		0.05	0.169	12.04 (0.90)	-0.83 (0.80)	-1.01 (0.80)	0.169 0.169	12.04 (0.90)	-0.83 (0.80)	-1.02 (0.80)	

**Table G.5:** Relative % error of ModSE (MCSE) for the difference in marginal RMST  $\Delta_\mu(20)$  for  $\gamma = 1.4$ ,  $\lambda = 0.003$  and  $n_{obs} = 10000$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised			
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest
0.1	0.5	-	0.083	3.74 (0.85)	-1.70 (0.80)	-2.01 (0.80)	0.078 0.112	5.31 2.95 (0.86) (0.85)	-1.04 -0.78 (0.81) (0.82)	-1.63 -1.28 (0.81)
		0.05	0.115	2.54 (0.85)	-1.04 (0.82)	-1.32 (0.82)				
	2	-	0.161	7.80 (0.88)	0.21 (0.82)	-0.35 (0.81)	0.166	6.41 (0.87)	-0.01 (0.81)	-0.56 (0.81)
		0.05	0.223	4.28 (0.85)	-0.69 (0.81)	-1.06 (0.81)	0.228	3.69 (0.85)	-0.77 (0.81)	-1.21 (0.81)
0.25	0.5	-	0.055	8.42 (0.88)	-0.12 (0.81)	-0.04 (0.81)	0.054	9.34 (0.88)	0.05 (0.81)	0.06 (0.81)
		0.05	0.079	4.80 (0.85)	-0.32 (0.81)	-0.27 (0.81)	0.078	5.10 (0.85)	-0.32 (0.81)	-0.21 (0.81)
	2	-	0.093	10.98 (0.90)	0.06 (0.81)	-0.00 (0.81)	0.094	9.69 (0.89)	-0.14 (0.81)	-0.22 (0.81)
		0.05	0.131	6.87 (0.86)	0.08 (0.81)	0.00 (0.81)	0.132	6.52 (0.86)	0.02 (0.81)	-0.00 (0.81)
0.5	0.5	-	0.057	7.94 (0.88)	-0.80 (0.81)	-0.56 (0.81)	0.057	7.85 (0.88)	-0.80 (0.81)	-0.64 (0.81)
		0.05	0.084	4.91 (0.85)	0.06 (0.81)	0.10 (0.81)	0.084	4.96 (0.85)	0.06 (0.81)	0.15 (0.81)
	2	-	0.068	13.53 (0.92)	1.09 (0.82)	1.27 (0.82)	0.068	13.64 (0.92)	1.10 (0.82)	1.36 (0.82)
		0.05	0.102	7.38 (0.87)	0.15 (0.81)	0.33 (0.81)	0.102	7.35 (0.87)	0.15 (0.81)	0.29 (0.81)

## G.4.2 Small Samples

The simulation study was designed to investigate the large sample properties of the proposed variance estimator. A sample size of 200 was also considered as an exploratory analysis. The exploratory results from the small sample simulation are shown here, excluding the estimates with the difficulties described in Section 5.6.3.

**Table G.6:** Relative % error of ModSE (MCSE) for the marginal log hazard ratio for  $\gamma = 0.7$ ,  $\lambda = 0.15$  and  $n_{obs} = 200$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised				
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest	
0.1	0.5	-	0.587 <sup>a</sup>	-19.37 (0.73)	34.73 (1.37)	-23.43 (1.04)	0.587	-19.34 (0.73)	29.52 (1.27)	-23.60 (1.05)	
		0.05	0.681	-20.20 (0.70)	68.43 (2.66)	-25.81 (0.82)	0.680	-19.87 (0.71)	50.97 (1.96)	-25.58 (0.82)	
	2	-	0.501	-29.16 (1.01)	135.36 (11.55)	-28.03 (2.46)	0.456	-27.27 (0.74)	17.02 (1.43)	-31.97 (1.03)	
		0.05	0.561 <sup>a</sup>	-22.34 (2.00)	126.04 (6.56)	-17.57 (5.51)	0.500	-21.24 (0.81)	34.85 (3.43)	-25.32 (1.85)	
	0.25	0.5	-	0.289	1.06 (0.88)	14.36 (1.04)	-14.97 (0.74)	0.291	0.76 (0.88)	14.01 (1.04)	-15.02 (0.74)
		0.05	0.339	0.01 (0.85)	12.09 (1.00)	-14.91 (0.73)	0.341	-0.04 (0.85)	11.89 (0.99)	-14.83 (0.73)	
0.5	2	-	0.273	-10.57 (0.84)	6.19 (1.17)	-23.44 (0.72)	0.266	-8.22 (0.87)	4.22 (1.11)	-22.01 (0.74)	
		0.05	0.292	-6.69 (0.84)	9.57 (1.10)	-20.04 (0.72)	0.286	-4.91 (0.86)	7.94 (1.07)	-19.00 (0.74)	
	0.5	0.5	-	0.189	10.94 (0.92)	7.61 (0.93)	-6.94 (0.78)	0.189	10.94 (0.92)	7.64 (0.93)	-6.91 (0.78)
		0.05	0.234	8.38 (0.89)	6.85 (0.89)	-6.92 (0.76)	0.234	8.39 (0.89)	6.88 (0.89)	-6.90 (0.76)	
	2	-	0.184	8.27 (0.92)	7.01 (0.95)	-9.18 (0.77)	0.183	8.27 (0.92)	6.89 (0.95)	-9.22 (0.77)	
		0.05	0.216	6.11 (0.89)	6.03 (0.92)	-10.24 (0.75)	0.216	6.14 (0.89)	5.98 (0.92)	-10.24 (0.75)	

<sup>a</sup>This was the empirical standard error for the robust and M-estimation variance estimators only. The empirical standard error corresponding to the bootstrap variance estimator was slightly different (when rounded to 3 decimal places). This was because a very small number of observations were removed due to the difficulties mentioned in Section 5.6.3.

**Table G.7:** Relative % error of ModSE (MCSE) for the marginal log hazard ratio for  $\gamma = 1.4$ ,  $\lambda = 0.003$  and  $n_{obs} = 200$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised			
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest
0.1	0.5	-	0.872 <sup>a</sup>	-19.68 (0.68)	233.33 (3.94)	-27.24 (0.72)	0.871	-19.66 (0.68)	185.00 (3.29)	-27.24 (0.72)
		0.05	1.050	-16.34 (0.73)	397.74 (4.78)	-25.43 (0.70)	1.043	-16.11 (0.74)	325.38 (4.07)	-25.14 (0.71)
	2	-	0.633	-18.13 (0.72)	41.15 (1.64)	-23.79 (0.85)	0.616	-17.22 (0.73)	34.35 (1.33)	-23.17 (0.88)
		0.05	0.834 <sup>a</sup>	-21.61 (0.68)	156.04 (4.39)	-28.35 (0.82)	0.806	-20.26 (0.69)	113.47 (2.83)	-27.38 (0.74)
	0.25	0.5	0.457	-2.42 (0.81)	39.46 (2.36)	-16.14 (0.71)	0.457	-2.45 (0.81)	34.81 (2.10)	-16.14 (0.71)
		0.05	0.673	-6.51 (0.79)	267.97 (4.71)	-16.34 (0.71)	0.673	-6.48 (0.79)	239.82 (4.30)	-16.32 (0.71)
0.5	2	-	0.329	2.70 (0.87)	13.90 (1.01)	-13.03 (0.74)	0.328	2.96 (0.87)	13.70 (1.00)	-12.93 (0.74)
		0.05	0.469	-3.09 (0.80)	13.49 (2.06)	-15.03 (0.71)	0.467	-2.77 (0.81)	11.68 (1.63)	-14.84 (0.71)
	0.5	0.5	0.351	3.30 (0.84)	5.07 (0.87)	-9.08 (0.74)	0.351	3.30 (0.84)	5.06 (0.87)	-9.08 (0.74)
		0.05	0.553	-3.78 (0.79)	84.60 (3.17)	-11.66 (0.73)	0.553	-3.78 (0.79)	83.51 (3.13)	-11.65 (0.73)
	2	-	0.287	5.63 (0.87)	6.97 (0.92)	-9.62 (0.75)	0.287	5.63 (0.87)	6.96 (0.92)	-9.62 (0.75)
		0.05	0.452	-3.37 (0.80)	44.36 (2.77)	-13.40 (0.72)	0.452	-3.36 (0.80)	43.79 (2.74)	-13.41 (0.72)

<sup>a</sup>This was the empirical standard error for the robust and M-estimation variance estimators only. The empirical standard error corresponding to the bootstrap variance estimator was slightly different (when rounded to 3 decimal places). This was because a very small number of observations were removed due to the difficulties mentioned in Section 5.6.3.

**Table G.8:** Relative % error of ModSE (MCSE) for the difference in marginal RMST  $\Delta_\mu(20)$  for  $\gamma = 0.7$ ,  $\lambda = 0.15$  and  $n_{obs} = 200$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised			
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest
0.1	0.5	-	2.571 <sup>a</sup>	-18.85 (0.72)	12.25 (0.99)	-26.78 (0.71)	2.540 <sup>a</sup>	-18.84 (0.72)	9.95 (0.97)	-27.02 (0.71)
		0.05	2.900	-18.53 (0.72)	8.44 (0.95)	-26.30 (0.80)	2.876	-18.17 (0.73)	7.03 (0.93)	-25.99 (0.80)
	2	-	2.521	-28.47 (0.73)	-4.94 (0.98)	-34.44 (0.94)	2.424	-25.79 (0.77)	-6.70 (0.96)	-32.55 (1.02)
		0.05	2.728	-20.57 (0.79)	5.24 (0.99)	-26.63 (0.98)	2.673 <sup>a</sup>	-19.08 (0.81)	4.52 (0.99)	-25.42 (1.09)
	0.25	0.5	1.371	1.22 (0.85)	7.93 (0.93)	-14.41 (0.72)	1.369	1.17 (0.85)	7.49 (0.93)	-14.45 (0.72)
		0.05	1.600	0.97 (0.85)	6.25 (0.91)	-13.67 (0.73)	1.601	0.91 (0.85)	5.94 (0.91)	-13.67 (0.73)
0.5	0.5	-	1.562	-8.26 (0.88)	-1.62 (1.00)	-21.81 (0.75)	1.537	-6.61 (0.90)	-1.52 (1.00)	-20.80 (0.77)
		0.05	1.657	-4.08 (0.88)	2.26 (0.97)	-18.28 (0.75)	1.645	-3.15 (0.90)	2.41 (0.97)	-17.76 (0.76)
	2	-	0.999	11.74 (0.92)	5.64 (0.88)	-6.77 (0.77)	0.999	11.75 (0.92)	5.64 (0.88)	-6.76 (0.77)
		0.05	1.243	8.62 (0.89)	4.67 (0.87)	-7.04 (0.76)	1.243	8.63 (0.89)	4.68 (0.87)	-7.02 (0.76)
	0.05	2	1.085	9.84 (0.95)	5.78 (0.94)	-8.33 (0.79)	1.085	9.83 (0.95)	5.70 (0.94)	-8.36 (0.79)
		0.05	1.279	7.69 (0.92)	4.49 (0.91)	-9.33 (0.77)	1.278	7.70 (0.92)	4.46 (0.91)	-9.33 (0.77)

<sup>a</sup>This was the empirical standard error for the robust and M-estimation variance estimators only. The empirical standard error corresponding to the bootstrap variance estimator was slightly different (when rounded to 3 decimal places). This was because a very small number of observations were removed due to the difficulties mentioned in Section 5.6.3.

**Table G.9:** Relative % error of ModSE (MCSE) for the difference in marginal RMST  $\Delta_\mu(20)$  for  $\gamma = 1.4$ ,  $\lambda = 0.003$  and  $n_{obs} = 200$

$\pi_Z$	$e^\beta$	$\lambda_C$	Unstabilised				Stabilised				
			Emp	Rob	Boot	Mest	Emp	Rob	Boot	Mest	
0.1	0.5	-	0.729	-7.97 (1.37)	30.52 (1.65)	-15.39 (2.00)	0.725	-7.63 (1.28)	33.87 (1.62)	-15.23 (1.69)	
		0.05	1.030	-11.87 (1.25)	20.88 (1.56)	-19.99 (1.57)	1.034 <sup>a</sup>	-11.21 (1.27)	21.00 (1.53)	-19.18 (1.67)	
	2	-	1.488	-16.68 (0.83)	18.40 (1.14)	-24.63 (0.86)	1.515	-18.61 (0.80)	16.55 (1.10)	-26.32 (0.84)	
		0.05	1.891	-19.47 (0.79)	10.92 (1.05)	-28.38 (0.79)	1.890	-18.84 (0.80)	10.42 (1.04)	-27.73 (0.78)	
	0.25	0.5	-	0.418	6.17 (1.03)	11.53 (1.12)	-8.23 (0.89)	0.416	6.85 (1.02)	12.29 (1.12)	-7.81 (0.87)
		0.05	0.584	3.02 (1.03)	8.39 (1.12)	-8.32 (0.90)	0.584	3.20 (1.04)	8.69 (1.13)	-8.19 (0.91)	
0.5	0.5	-	0.751	2.52 (0.91)	10.61 (1.02)	-13.67 (0.77)	0.759	1.52 (0.90)	9.94 (1.00)	-13.91 (0.77)	
		0.05	1.023	-2.84 (0.85)	4.87 (0.94)	-15.74 (0.74)	1.030	-2.97 (0.85)	4.62 (0.94)	-15.68 (0.74)	
	2	-	0.437	5.70 (1.12)	6.89 (1.18)	-8.69 (0.93)	0.437	5.73 (1.12)	6.87 (1.18)	-8.66 (0.93)	
		0.05	0.634	0.39 (1.10)	2.27 (1.14)	-10.53 (0.93)	0.634	0.42 (1.10)	2.28 (1.14)	-10.50 (0.93)	
	0.05	-	0.528	10.87 (1.04)	8.31 (1.02)	-5.98 (0.86)	0.528	10.85 (1.04)	8.35 (1.02)	-5.95 (0.86)	
		0.05	0.782	4.26 (0.98)	4.18 (1.01)	-7.92 (0.85)	0.782	4.24 (0.98)	4.19 (1.01)	-7.91 (0.85)	

<sup>a</sup>This was the empirical standard error for the robust and M-estimation variance estimators only. The empirical standard error corresponding to the bootstrap variance estimator was slightly different (when rounded to 3 decimal places). This was because a very small number of observations were removed due to the difficulties mentioned in Section 5.6.3.

# Appendix H

---

## Additional Material for Chapter 6

---

This appendix provides additional material for Chapter 6: Software development for the closed-form variance estimator for inverse probability weighted parametric survival models. The appendix includes information on the variables used in the analysis of the STD and ACTG175 datasets. The code using `stipw` is given in Sections 6.5.1 and 6.5.2, respectively, with the corresponding results given in Sections 5.7.2 and 5.7.3.

### H.1 Data Variable Definitions

**Table H.1:** STD dataset variable description, as described in R package KMsurv [5]

Variable name	Variable description	O/F	Inc	Notes
time	Time to reinfection (days)	O	No	years used instead
years	Time to reinfection (years)	F	Yes	Time-to-event variable
rinfct	Reinfection: 0 = censored 1 = reinfection	O	Yes	Event indicator variable
race	Race: 0 = W = White 1 = B = Black	O	Yes	Exposure variable
marital	Marital status D = divorced/separated M = married S = single	O	No	DVs used instead
marital2	Single <sup>1</sup>	F	Yes	Confounder (DV)
marital3	Divorced/separated <sup>1</sup>	F	Yes	Confounder (DV)
age	Age at initial infection	O	Yes	Confounder
yschool	Years of schooling	O	Yes	Confounder
iinfct	Type of initial infection: 1 = gonorrhea 2 = chlamydia 3 = both	O	No	DVs used instead
iinfct2	Initial infection: chlamydia <sup>1</sup>	F	Yes	Confounder (DV)
iinfct3	Initial infection: both <sup>1</sup>	F	Yes	Confounder (DV)
npartner	Number of sexual partners in the last 30 days	O	No	DVs used instead
npartner1	1 sexual partner <sup>1</sup>	F	Yes	Confounder (DV)
npartner2	2 sexual partners <sup>1</sup>	F	Yes	Confounder (DV)
npartner3	>=3 sexual partners <sup>1</sup>	F	Yes	Confounder (DV)
os12m	Oral sex <sup>1,2</sup>	O	Yes	Confounder
rs12m	Rectal sex <sup>1,2</sup>	O	Yes	Confounder
condom	Condom use 1 = always 2 = sometimes 3 = never	O	No	DVs used instead
condom2	Condom use: sometimes <sup>1</sup>	F	Yes	Confounder (DV)
condom3	Condom use: never <sup>1</sup>	F	Yes	Confounder (DV)
abdpain	Abdominal pain <sup>1</sup>	O	Yes	Confounder
discharge	Sign of discharge <sup>1</sup>	O	Yes	Confounder
dysuria	Sign of dysuria <sup>1</sup>	O	Yes	Confounder
itch	Sign of itch <sup>1</sup>	O	Yes	Confounder
lesion	Sign of lesion <sup>1</sup>	O	Yes	Confounder
rash	Sign of rash <sup>1</sup>	O	Yes	Confounder
lymph	Sign of lymph involvement <sup>1</sup>	O	Yes	Confounder

O = Original, F = Formatted, Inc = Included in the analysis, DV = Dummy variable. <sup>1</sup>1 = yes, 0 = no. <sup>2</sup>within past 12 months.

**Table H.2:** ACTG175 dataset variable description, as described in R package `speff2trial` [6]

Variable name	Variable description	O/F	Inc	Notes
days	Number of days until the first occurrence of: (i) a decline in CD4 T cell count of at least 50 (ii) an event indicating progression to AIDS (iii) death	O	No	years used instead
years	Time to event in years (see above)	F	Yes	Time-to-event variable
cens	Indicator of observing the event: 0 = censored 1 = event	O	Yes	Event indicator variable
arms	Treatment arm: 0 = zidovudine 1 = zidovudine and didanosine 2 = zidovudine and zalcitabine 3 = didanosine	O	No	trt used instead
trt	Treatment arm: 0 = monotherapy (arms 0 and 3) 1 = combination therapy (arms 1 and 2)	F	Yes	Treatment variable
age	Age in years at baseline	O	Yes	Confounder
wtkg	Weight in kilograms at baseline	O	Yes	Confounder
hemo	Hemophilia <sup>1</sup>	O	Yes	Confounder
homo	Homosexual activity <sup>1</sup>	O	Yes	Confounder
drugs	History of intravenous drug use <sup>1</sup>	O	Yes	Confounder
karnof	Karnofsky score (on a scale of 0-100)	O	Yes	Confounder
oprior	Non-zidovudine antiretroviral therapy prior to initiation of study treatment <sup>1</sup>	O	Yes	Confounder

O = Original, F = Formatted, Inc = Included in the analysis, DV = Dummy variable. <sup>1</sup>1 = yes, 0 = no.

**Table H.3:** ACTG175 dataset variable description, as described in R package `speff2trial` [6] (continued)

Variable name	Variable description	O/F	Inc	Notes
z30	Zidovudine use in the 30 days prior to treatment initiation <sup>1</sup>	O	Yes	Confounder
zprior	Zidovudine use prior to treatment initiation <sup>1</sup>	O	No	Singular
preanti	Number of days of previously received antiretroviral therapy	O	Yes	Confounder
race	Race: 0 = White 1 = Non-White	O	Yes	Confounder
gender	Gender: 0 = female 1 = male	O	Yes	Confounder
str2	Antiretroviral history: 0 = naïve 1 = experienced	O	No	Collinear with strat
strat	Antiretroviral history stratification: 1 = antiretroviral naïve 2 = >1 but $\leqslant$ 52 weeks of prior antiretroviral therapy 3 = >52 weeks of prior antiretroviral therapy	O	No	DVs used instead
strat2	>1 but $\leqslant$ 52 weeks of prior antiretroviral therapy <sup>1</sup>	F	Yes	Confounder (DV)
strat3	>52 weeks of prior antiretroviral therapy <sup>1</sup>	F	Yes	Confounder (DV)
symptom	Symptomatic indicator: 0 = asymptomatic 1 = symptomatic	O	Yes	Confounder
cd40	CD4 T cell count at baseline	O	Yes	Confounder
cd80	CD8 T cell count at baseline	O	Yes	Confounder

# Appendix I

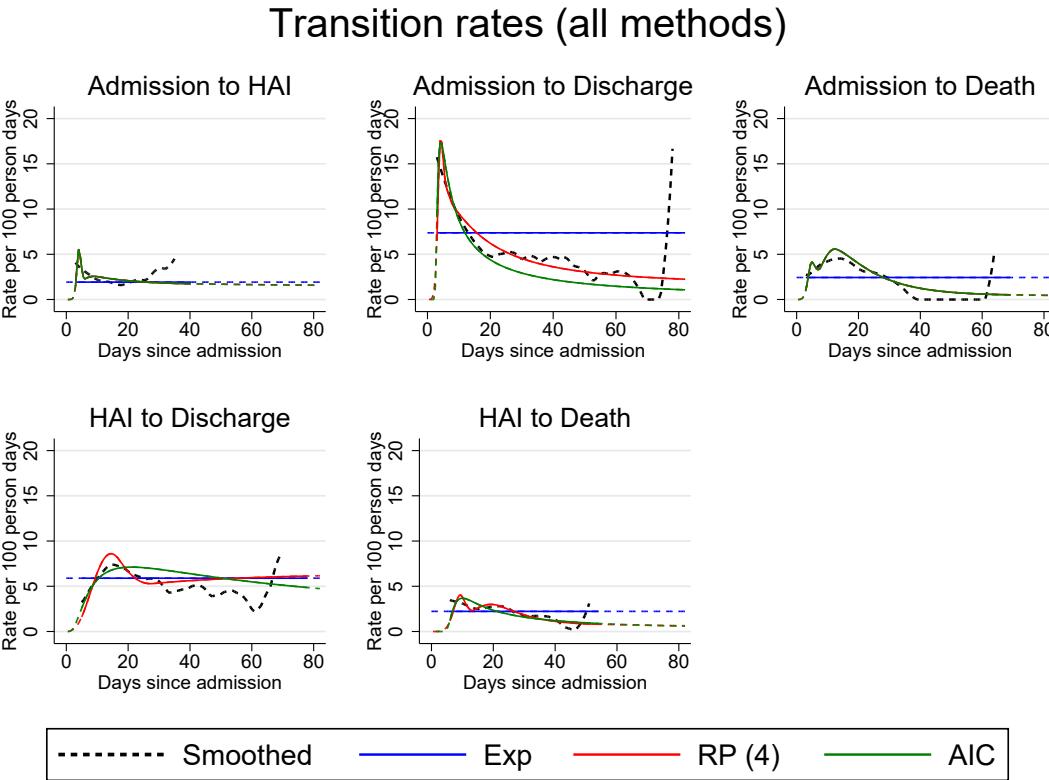
---

## Additional Material for Chapter 7

---

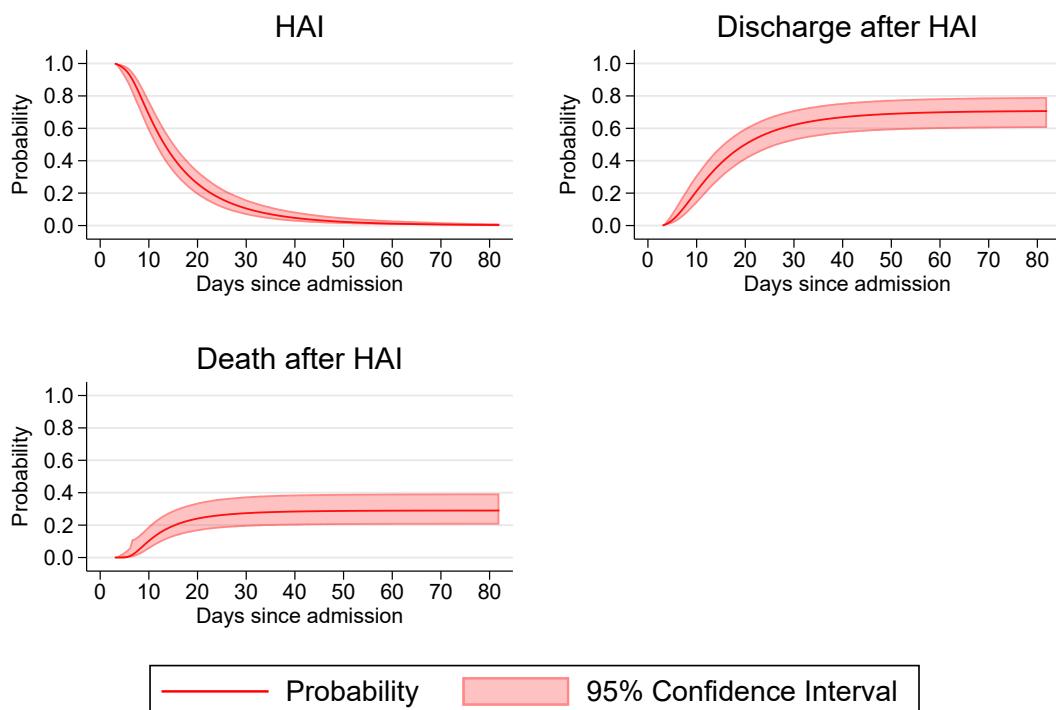
This appendix provides additional figures for Chapter 7: Multistate model application to the hospital acquired infection dataset and corresponding software development. The additional figures given here correspond to the results of the analysis given in Section 7.5.

## I.1 Additional Figures



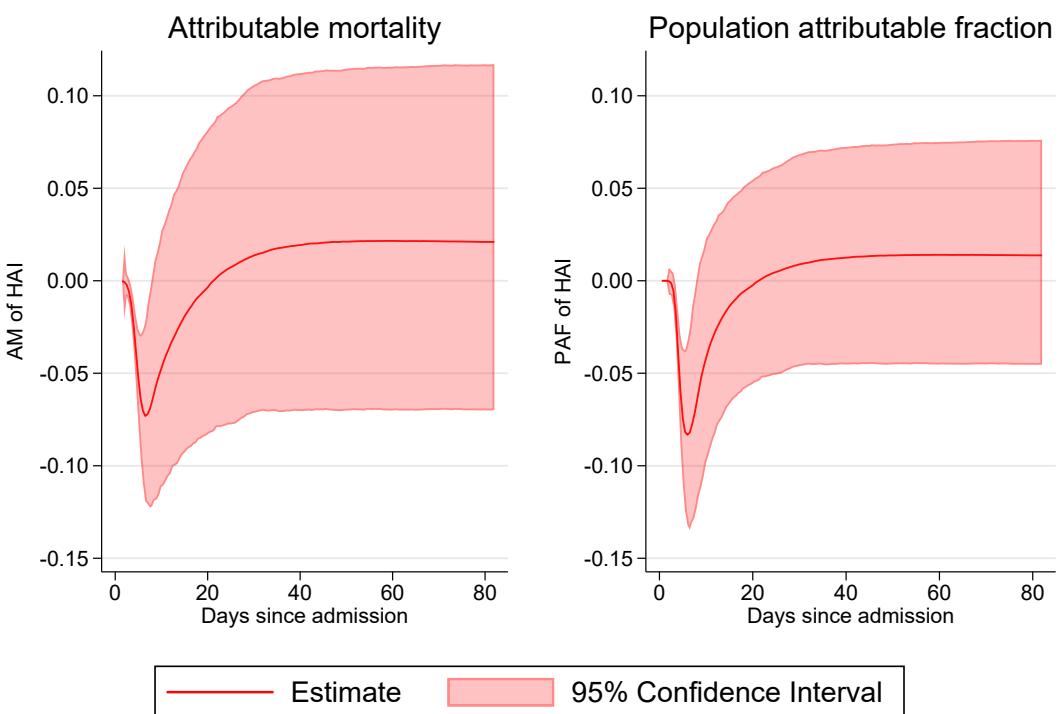
**Figure I.1:** Transition rates for the different approaches for the HAI data. The Epanechnikov kernel was used for smoothed non-parametric estimates. Estimates from the parametric approaches were defined from the time of the first event until the last event for each transition by a solid line and were extrapolated to cover the interval  $[0, 82]$  by a dashed line. The smoothed non-parametric estimates were truncated up to 8 days before the last event, as it was not believed that the transition rates increased so drastically at the time of the last event

### Transition probabilities: state 2, time 3 (AIC model)



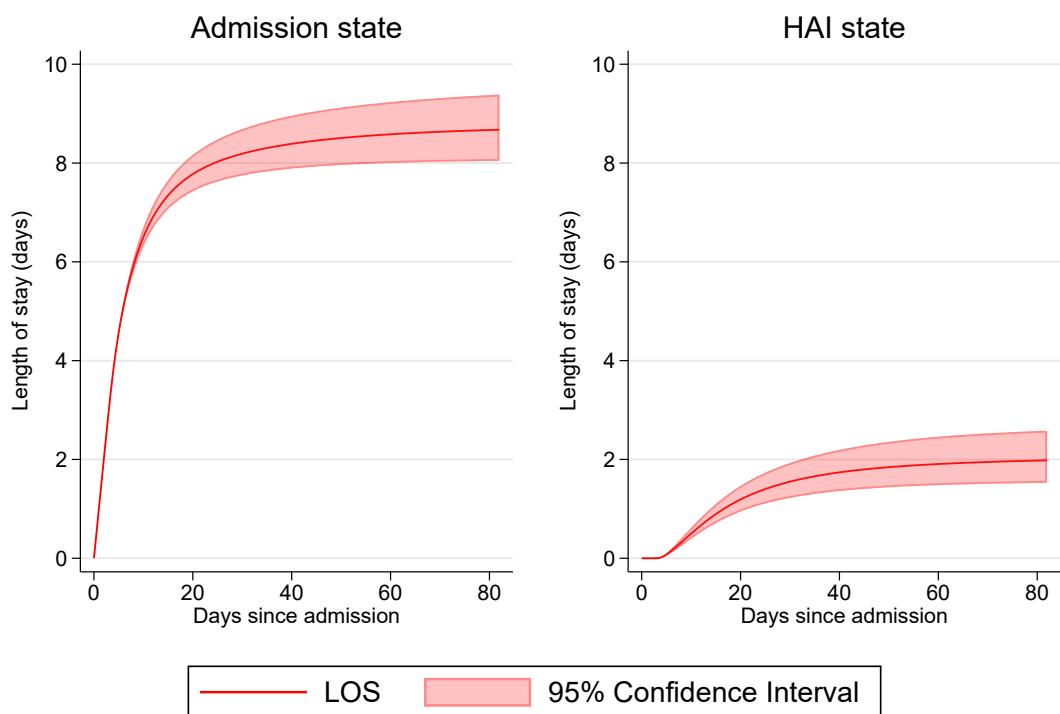
**Figure I.2:** Transition probabilities from state 2 at time 3 to the relevant states for the “AIC” model with 95% confidence intervals for the HAI data

### AM & PAF (AIC model)



**Figure I.3:** Attributable mortality (AM) and population attributable fraction (PAF) of HAIs for the “AIC” model with 95% confidence intervals for the HAI data

### Length of stay: state 1, time 0 (AIC model)



**Figure I.4:** Length of stay in hospital without (state 1, left panel) and with (state 2, right panel) a HAI starting from state 1 at time 0 for the “AIC” model with 95% confidence intervals for the HAI data

---

## Bibliography

---

- [1] H. Putter, J. van der Hage, G. H. de Bock, R. Elgalta, and C. J. H. van de Velde. Estimation and prediction in a multi-state model for breast cancer. *Biometrical Journal: Journal of Mathematical Methods in Biosciences*, 48(3):366–380, 2006.
- [2] D. Hajage, G. Chauvet, L. Belin, A. Lafourcade, F. Tubach, and Y. De Rycke. Closed-form variance estimator for weighted propensity score estimators with survival outcome. *Biometrical Journal*, 60(6):1151–1163, 2018.
- [3] K. Bogaerts, A. Komárek, and E. Lesaffre. *Survival analysis with interval-censored data: a practical approach with examples in R, SAS, and BUGS*. Chapman and Hall/CRC, 2017.
- [4] T. P. Morris, I. R. White, and M. J. Crowther. Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11):2074–2102, 2019.
- [5] J. Yan. *KMsurv: data sets from Klein and Moeschberger (1997), survival analysis*, 2012. Original by Klein and Moeschberger. R package, version 0.1-5. Available from CRAN at <https://CRAN.R-project.org/package=KMsurv>.
- [6] M. Juraska, P. B. Gilbert, X. Lu, M. Zhang, M. Davidian, and A. A. Tsiatis. *speff2trial: semiparametric efficient estimation for a two-sample treatment effect*, 2012. R package, version 1.0.4. Available from CRAN at <https://CRAN.R-project.org/package=speff2trial>.
- [7] P. Schober and T. R. Vetter. Survival analysis and interpretation of time-to-event data: the tortoise and the hare. *Anesthesia and Analgesia*, 127(3):792–798, 2018.
- [8] P. Royston and M. K. B. Parmar. Flexible parametric proportional-hazards and proportional-odds models for censored survival data, with application to prognostic modelling and estimation of treatment effects. *Statistics in Medicine*, 21(15):2175–2197, 2002.
- [9] P. R. Rosenbaum and D. B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.
- [10] P. R. Rosenbaum and D. B. Rubin. Constructing a control group using multivariate matched sampling methods that incorporate the propensity score. *The American Statistician*, 39(1):33–38, 1985.
- [11] P. R. Rosenbaum and D. B. Rubin. Reducing bias in observational studies using subclassification on the propensity score. *Journal of the American Statistical Association*, 79(387):516–524, 1984.

- [12] P. R. Rosenbaum. Model-based direct adjustment. *Journal of the American Statistical Association*, 82(398):387–394, 1987.
- [13] R. B. D’Agostino Jr. Propensity score methods for bias reduction in the comparison of a treatment to a non-randomized control group. *Statistics in Medicine*, 17(19):2265–2281, 1998.
- [14] P. C. Austin. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate Behavioral Research*, 46(3):399–424, 2011.
- [15] M. A. Hernán and J. M. Robins. *Causal inference: what if*. Chapman and Hall/CRC, 2020.
- [16] G. W. Imbens. Nonparametric estimation of average treatment effects under exogeneity: a review. *The Review of Economics and Statistics*, 86(1):4–29, 2004.
- [17] P. C. Austin. A tutorial and case study in propensity score analysis: an application to estimating the effect of in-hospital smoking cessation counseling on mortality. *Multivariate Behavioral Research*, 46(1):119–151, 2011.
- [18] J. M. Robins, M. A. Hernán, and B. Brumback. Marginal structural models and causal inference in epidemiology. *Epidemiology*, 11(5):550–560, 2000.
- [19] P. C. Austin. Variance estimation when using inverse probability of treatment weighting (IPTW) with survival analysis. *Statistics in Medicine*, 35(30):5642–5655, 2016.
- [20] H. Mao, L. Li, W. Yang, and Y. Shen. On the propensity score weighting analysis with survival outcome: estimands, estimation, and inference. *Statistics in Medicine*, 37(26):3745–3763, 2018.
- [21] D. Shu, J. G. Young, S. Toh, and R. Wang. Variance estimation in inverse probability weighted Cox models. *Biometrics*, 77(3):1101–1117, 2021.
- [22] M. J. Crowther and P. C. Lambert. Parametric multistate survival models: flexible modelling allowing transition-specific distributions with application to estimating clinically useful measures of effect differences. *Statistics in Medicine*, 36(29):4719–4742, 2017.
- [23] M. von Cube, M. Schumacher, and M. Wolkewitz. Basic parametric analysis for a multi-state model in hospital epidemiology. *BMC Medical Research Methodology*, 17:111, 2017.
- [24] S. M. Hammer, D. A. Katzenstein, M. D. Hughes, H. Gundacker, R. T. Schooley, R. H. Haubrich, et al. A trial comparing nucleoside monotherapy with combination therapy in HIV-infected adults with CD4 cell counts from 200 to 500 per cubic millimeter. *New England Journal of Medicine*, 335(15):1081–1090, 1996.

- [25] G. F. Beadle, B. Silver, L. Botnick, S. Hellman, and J. R. Harris. Cosmetic results following primary radiation therapy for early breast cancer. *Cancer*, 54(12):2911–2918, 1984.
- [26] G. F. Beadle, S. Come, I. C. Henderson, B. Silver, S. Hellman, and J. R. Harris. The effect of adjuvant chemotherapy on the cosmetic results after primary radiation treatment for early stage breast cancer. *International Journal of Radiation Oncology Biology Physics*, 10(11):2131–2137, 1984.
- [27] A. Komárek, K. Bogaerts, and E. Lesaffre. *icensBKL: accompanion to the book on interval censoring by Bogaerts, Komarek, and Lesaffre*, 2020. R package, version 1.2. Available from CRAN at <https://CRAN.R-project.org/package=icensBKL>.
- [28] C. Pan, B. Cai, L. Wang, and X. Lin. *ICBayes: Bayesian semiparametric models for interval-censored data*, 2020. R package, version 1.2. Available from CRAN at <https://CRAN.R-project.org/package=ICBayes>.
- [29] M. P. Fay. *interval: weighted logrank tests and NPMLE for interval censored data*, 2021. R package, version 1.1-0.8. Available from CRAN at <https://CRAN.R-project.org/package=interval>.
- [30] J. Beyersmann, P. Gastmeier, H. Grundmann, S. Bärwolff, C. Geffers, M. Behnke, et al. Use of multistate models to assess prolongation of intensive care unit stay due to nosocomial infection. *Infection Control & Hospital Epidemiology*, 27(5):493–499, 2006.
- [31] A. Allignol, M. Schumacher, and J. Beyersmann. Empirical transition matrix of multi-state models: the etm package. *Journal of Statistical Software*, 38(4): 1–15, 2011.
- [32] A. F. Connors, T. Speroff, N. V. Dawson, C. Thomas, F. E. Harrell, D. Wagner, et al. The effectiveness of right heart catheterization in the initial care of critically ill patients. *JAMA*, 276(11):889–897, 1996.
- [33] F. E. Harrell Jr and C. Dupont. *Hmisc: Harrell miscellaneous*, 2021. R package, version 4.6-0. Available from CRAN at <https://CRAN.R-project.org/package=Hmisc>.
- [34] J. P. Klein and M. L. Moeschberger. *Survival analysis: techniques for censored and truncated data*. Springer, 1997.
- [35] M. Hill, P. C. Lambert, and M. J. Crowther. Relaxing the assumption of constant transition rates in a multi-state model in hospital epidemiology. *BMC Medical Research Methodology*, 21:16, 2021.
- [36] D. Collett. *Modelling survival data in medical research (3rd edition)*. Chapman and Hall/CRC, 3rd edition, 2015.
- [37] O. Aalen, Ø. Borgan, and H. Gjessing. *Survival and event history analysis: a process point of view*. Springer, 2008.

- [38] J. P. Klein, H. C. van Houwelingen, J. G. Ibrahim, and T. H. Scheike. *Handbook of survival analysis*. Chapman and Hall/CRC, 2014.
- [39] P. K. Andersen and M. Pohar Perme. Pseudo-observations in survival analysis. *Statistical Methods in Medical Research*, 19(1):71–99, 2010.
- [40] P. Royston and M. K. B. Parmar. The use of restricted mean survival time to estimate the treatment effect in randomized clinical trials when the proportional hazards assumption is in doubt. *Statistics in Medicine*, 30(19):2409–2421, 2011.
- [41] P. Royston and M. K. B. Parmar. Restricted mean survival time: an alternative to the hazard ratio for the design and analysis of randomized trials with a time-to-event outcome. *BMC Medical Research Methodology*, 13:152, 2013.
- [42] P. C. Lambert and P. Royston. Further development of flexible parametric models for survival analysis. *The Stata Journal*, 9(2):265–290, 2009.
- [43] M. J. Rutherford, M. J. Crowther, and P. C. Lambert. The use of restricted cubic splines to approximate complex hazard functions in the analysis of time-to-event data: a simulation study. *Journal of Statistical Computation and Simulation*, 85(4):777–793, 2015.
- [44] E. Syriopoulou, S. I. Mozumder, M. J. Rutherford, and P. C. Lambert. Robustness of individual and marginal model-based estimates: a sensitivity analysis of flexible parametric models. *Cancer Epidemiology*, 58:17–24, 2019.
- [45] H. Bower, M. J. Crowther, M. J. Rutherford, T. M. L. Andersson, M. Clements, X. R. Liu, et al. Capturing simple and complex time-dependent effects using flexible parametric survival models: a simulation study. *Communications in Statistics - Simulation and Computation*, 50(11):3777–3793, 2021.
- [46] D. R. Cox and D. Oakes. *Analysis of survival data*. Chapman and Hall/CRC, 1984.
- [47] M. J. Crowther, P. Royston, and M. Clements. A flexible parametric accelerated failure time model. *arXiv preprint*, arXiv:2006.06807, 2020. Preprint at <https://arxiv.org/abs/2006.06807>.
- [48] M. Pang, R. W. Platt, T. Schuster, and M. Abrahamowicz. Spline-based accelerated failure time model. *Statistics in Medicine*, 40(2):481–497, 2021.
- [49] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282):457–481, 1958.
- [50] M. Greenwood. *The errors of sampling of the survivorship tables*. Number 33, Appendix 1, HMSO, 1926.
- [51] W. Nelson. Hazard plotting for incomplete failure data. *Journal of Quality Technology*, 1(1):27–52, 1969.

- [52] W. Nelson. Theory and applications of hazard plotting for censored failure data. *Technometrics*, 14(4):945–966, 1972.
- [53] O. Aalen. Nonparametric inference for a family of counting processes. *The Annals of Statistics*, 6(4):701–726, 1978.
- [54] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.
- [55] N. Breslow. Covariance analysis of censored survival data. *Biometrics*, 30(1):89–99, 1974.
- [56] B. Efron. The efficiency of Cox’s likelihood function for censored data. *Journal of the American Statistical Association*, 72(359):557–565, 1977.
- [57] J. D. Kalbfleisch and R. L. Prentice. Marginal likelihoods based on Cox’s regression and life model. *Biometrika*, 60(2):267–278, 1973.
- [58] D. R. Cox and E. J. Snell. A general definition of residuals. *Journal of the Royal Statistical Society: Series B (Methodological)*, 30(2):248–265, 1968.
- [59] W. E. Barlow and R. L. Prentice. Residuals for relative risk regression. *Biometrika*, 75(1):65–74, 1988.
- [60] T. M. Therneau, P. M. Grambsch, and T. R. Fleming. Martingale-based residuals for survival models. *Biometrika*, 77(1):147–160, 1990.
- [61] D. Schoenfeld. Partial residuals for the proportional hazards regression model. *Biometrika*, 69(1):239–241, 1982.
- [62] P. M. Grambsch and T. M. Therneau. Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81(3):515–526, 1994.
- [63] StataCorp. *Stata 17 Base Reference Manual*. Stata Press, 2021.
- [64] T. M. Therneau and P. M. Grambsch. *Modelling survival data: extending the Cox model*. Springer, 2000.
- [65] T. R. Fleming and D. P. Harrington. *Counting processes and survival analysis*. John Wiley & Sons, 2005.
- [66] P. M. Odell, K. M. Anderson, and R. B. D’Agostino. Maximum likelihood estimation for interval-censored data using a Weibull-based accelerated failure time model. *Biometrics*, 48(3):951–959, 1992.
- [67] J. C. Lindsey and L. M. Ryan. Methods for interval-censored data. *Statistics in Medicine*, 17(2):219–238, 1998.
- [68] G. Gómez, M. L. Calle, R. Oller, and K. Langohr. Tutorial on methods for interval-censored data and their implementation in R. *Statistical Modelling*, 9(4):259–297, 2009.

- [69] E. Lesaffre, A. Komárek, and D. Declerck. An overview of methods for interval-censored data with an emphasis on applications in dentistry. *Statistical Methods in Medical Research*, 14(6):539–552, 2005.
- [70] B. W. Turnbull. The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 38(3):290–295, 1976.
- [71] R. Peto and J. Peto. Asymptotically efficient rank invariant test procedures. *Journal of the Royal Statistical Society: Series A (General)*, 135(2):185–198, 1972.
- [72] P. Groeneboom and J. A. Wellner. *Information bounds and nonparametric maximum likelihood estimation*. Birkhäuser-Verlag, 1992.
- [73] J. A. Wellner and Y. Zhan. A hybrid algorithm for computation of the nonparametric maximum likelihood estimator from censored data. *Journal of the American Statistical Association*, 92(439):945–959, 1997.
- [74] D. M. Finkelstein. A proportional hazards model for interval-censored failure time data. *Biometrics*, 42(4):845–854, 1986.
- [75] E. Goetghebeur and L. Ryan. Semiparametric regression analysis of interval-censored data. *Biometrics*, 56(4):1139–1144, 2000.
- [76] C. P. Farrington. Interval censored survival data: a generalized linear modelling approach. *Statistics in Medicine*, 15(3):283–292, 1996.
- [77] W. Pan. Extending the iterative convex minorant algorithm to the Cox model for interval-censored data. *Journal of Computational and Graphical Statistics*, 8(1):109–120, 1999.
- [78] D. B. Rubin. *Multiple imputation for nonresponse in surveys*. John Wiley & Sons, New York, 1987.
- [79] W. Pan. A multiple imputation approach to Cox regression with interval-censored data. *Biometrics*, 56(1):199–203, 2000.
- [80] W. B. Goggins, D. M. Finkelstein, D. A. Schoenfeld, and A. M. Zaslavsky. A Markov chain Monte Carlo EM algorithm for analyzing interval-censored data under the Cox proportional hazards model. *Biometrics*, 54(4):1498–1507, 1998.
- [81] G. A. Satten. Rank-based inference in the proportional hazards model for interval censored data. *Biometrika*, 83(2):355–370, 1996.
- [82] E. J. Williamson, A. Forbes, and I. R. White. Variance reduction in randomised trials by inverse probability weighting using the propensity score. *Statistics in Medicine*, 33(5):721–737, 2014.
- [83] P. Y. Chen and A. A. Tsiatis. Causal inference on the difference of the restricted mean lifetime between two groups. *Biometrics*, 57(4):1030–1038, 2001.

- [84] M. A. Hernán. The hazards of hazard ratios. *Epidemiology*, 21(1):13–15, 2010.
- [85] S. Greenland. Absence of confounding does not correspond to collapsibility of the rate ratio or rate difference. *Epidemiology*, 7(5):498–501, 1996.
- [86] S. Greenland. Interpretation and choice of effect measures in epidemiologic analyses. *American Journal of Epidemiology*, 125(5):761–768, 1987.
- [87] S. R. Cole and M. A. Hernán. Constructing inverse probability weights for marginal structural models. *American Journal of Epidemiology*, 168(6):656–664, 2008.
- [88] D. B. Rubin. Randomization analysis of experimental data: the Fisher randomization test comment. *Journal of the American Statistical Association*, 75(371):591–593, 1980.
- [89] T. J. VanderWeele, M. A. Hernán, and J. M. Robins. Causal directed acyclic graphs and the direction of unmeasured confounding bias. *Epidemiology*, 19(5):720–728, 2008.
- [90] J. M. Robins and S. Greenland. The role of model selection in causal inference from nonexperimental data. *American Journal of Epidemiology*, 123(3):392–402, 1986.
- [91] H. Bang and J. M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.
- [92] G. W. Imbens. The role of the propensity score in estimating dose-response functions. *Biometrika*, 87(3):706–710, 2000.
- [93] K. Imai and D. A. Van Dyk. Causal inference with general treatment regimes: generalizing the propensity score. *Journal of the American Statistical Association*, 99(467):854–866, 2004.
- [94] P. C. Austin and E. A. Stuart. Moving towards best practice when using inverse probability of treatment weighting (IPTW) using the propensity score to estimate causal treatment effects in observational studies. *Statistics in Medicine*, 34(28):3661–3679, 2015.
- [95] B. K. Lee, J. Lessler, and E. A. Stuart. Weight trimming and propensity score weighting. *PLoS ONE*, 6(3):e18174, 2011.
- [96] L. Li and T. Greene. A weighting analogue to pair matching in propensity score analysis. *The International Journal of Biostatistics*, 9(2):215–234, 2013.
- [97] F. Li, K. L. Morgan, and A. M. Zaslavsky. Balancing covariates via propensity score weighting. *Journal of the American Statistical Association*, 113(521):390–400, 2018.
- [98] A. Breskin, S. R. Cole, and D. Westreich. Exploring the subtleties of inverse probability weighting and marginal structural models. *Epidemiology*, 29(3):352–355, 2018.

- [99] M. Fiocco, H. Putter, and H. C. van Houwelingen. Reduced-rank proportional hazards regression and simulation-based prediction for multi-state models. *Statistics in Medicine*, 27(21):4340–4358, 2008.
- [100] M. K. Grand and H. Putter. Regression models for expected length of stay. *Statistics in Medicine*, 35(7):1178–1192, 2016.
- [101] C. H. Jackson. *Flexible parametric multi-state modelling with flexsurv*. R package vignette, accessed on 14/04/2022 at <https://cran.r-project.org/web/packages/flexsurv/vignettes/multistate.pdf>.
- [102] P. K. Andersen and N. Keiding. Multi-state models for event history analysis. *Statistical Methods in Medical Research*, 11(2):91–115, 2002.
- [103] P. K. Andersen, Ø. Borgan, R. D. Gill, and N. Keiding. *Statistical models based on counting processes*. Springer, 1993.
- [104] O. O. Aalen and S. Johansen. An empirical transition matrix for non-homogeneous Markov chains based on censored observations. *Scandinavian Journal of Statistics*, 5(3):141–150, 1978.
- [105] D. R. Cox and H. D. Miller. *The theory of stochastic processes*. Chapman and Hall/CRC, 1965.
- [106] A. C. Titman. Flexible nonhomogeneous Markov models for panel observed data. *Biometrics*, 67(3):780–787, 2011.
- [107] C. Jackson. Multi-state models for panel data: the msm package for R. *Journal of Statistical Software*, 38(8):1–28, 2011.
- [108] S. Greenland and W. D. Finkle. A critical look at methods for handling missing covariates in epidemiologic regression analyses. *American Journal of Epidemiology*, 142(12):1255–1264, 1995.
- [109] A. R. T. Donders, G. J. M. G. van der Heijden, T. Stijnen, and K. G. M. Moons. A gentle introduction to imputation of missing values. *Journal of Clinical Epidemiology*, 59(10):1087–1091, 2006.
- [110] J. D. Dziura, L. A. Post, Q. Zhao, Z. Fu, and P. Peduzzi. Strategies for dealing with missing data in clinical trials: from design to analysis. *The Yale Journal of Biology and Medicine*, 86(3):343–358, 2013.
- [111] Z. Zhang and J. Sun. Interval censoring. *Statistical Methods in Medical Research*, 19(1):53–70, 2010.
- [112] K. S. Panageas, L. Ben-Porat, M. N. Dickler, P. B. Chapman, and D. Schrag. When you look matters: the effect of assessment schedule on progression-free survival. *Journal of the National Cancer Institute*, 99(6):428–432, 2007.
- [113] Y. Qi, K. L. Allen Ziegler, S. L. Hillman, M. W. Redman, S. E. Schild, D. R. Gandara, et al. Impact of disease progression date determination on progression-free survival estimates in advanced lung cancer. *Cancer*, 118(21):5358–5365, 2012.

- [114] X. Sun and C. Chen. Comparison of Finkelstein's method with the conventional approach for interval-censored data analysis. *Statistics in Biopharmaceutical Research*, 2(1):97–108, 2010.
- [115] S. Banerjee, K. N. Moore, N. Colombo, G. Scambia, B. G. Kim, A. Oaknin, et al. Maintenance olaparib for patients with newly diagnosed advanced ovarian cancer and a BRCA mutation (SOLO1/GOG 3004): 5-year follow-up of a randomised, double-blind, placebo-controlled, phase 3 trial. *The Lancet Oncology*, 22(12):1721–1731, 2021.
- [116] J. W. Valle, A. Vogel, C. S. Denlinger, A. R. He, L. Y. Bai, R. Orlova, et al. Addition of ramucirumab or merestinib to standard first-line chemotherapy for locally advanced or metastatic biliary tract cancer: a randomised, double-blind, multicentre, phase 2 study. *The Lancet Oncology*, 22(10):1468–1482, 2021.
- [117] Y. Yang, S. Qu, J. Li, C. Hu, M. Xu, W. Li, et al. Camrelizumab versus placebo in combination with gemcitabine and cisplatin as first-line treatment for recurrent or metastatic nasopharyngeal carcinoma (CAPTAIN-1st): a multicentre, randomised, double-blind, phase 3 trial. *The Lancet Oncology*, 22(8):1162–1174, 2021.
- [118] C. Zhou, Z. Wang, Y. Sun, L. Cao, Z. Ma, R. Wu, et al. Sugemalimab versus placebo, in combination with platinum-based chemotherapy, as first-line treatment of metastatic non-small-cell lung cancer (GEMSTONE-302): interim and final analyses of a double-blind, randomised, phase 3 clinical trial. *The Lancet Oncology*, 23(2):220–233, 2022.
- [119] J. Esbjörnsson, F. Måansson, A. Kvist, Z. J. da Silva, S. Andersson, E. M. Fenyö, et al. Long-term follow-up of HIV-2-related AIDS and mortality in Guinea-Bissau: a prospective open cohort study. *The Lancet HIV*, 6(1):e25–e31, 2019.
- [120] C. M. McDonald, E. K. Henricson, R. T. Abresch, T. Duong, N. C. Joyce, F. Hu, et al. Long-term effects of glucocorticoids on function, quality of life, and survival in patients with Duchenne muscular dystrophy: a prospective cohort study. *The Lancet*, 391(10119):451–461, 2018.
- [121] L. Zeng, R. J. Cook, L. Wen, and A. Boruvka. Bias in progression-free survival analysis due to intermittent assessment of progression. *Statistics in Medicine*, 34(24):3181–3193, 2015.
- [122] J. Harezlak and W. Tu. Estimation of survival functions in interval and right censored data using STD behavioural diaries. *Statistics in Medicine*, 25(23):4053–4064, 2006.
- [123] G. MacKenzie and D. Peng. Interval-censored parametric regression survival models and the analysis of longitudinal trials. *Statistics in Medicine*, 32(16):2804–2822, 2013.

- [124] H. Støvring and I. S. Kristiansen. Simple parametric survival analysis with anonymized register data: a cohort study with truncated and interval censored event and censoring times. *BMC Research Notes*, 4:308, 2011.
- [125] X. Song and S. Ma. Multiple augmentation for interval-censored data with measurement error. *Statistics in Medicine*, 27(16):3178–3190, 2008.
- [126] W. Fu and J. S. Simonoff. Survival trees for interval-censored survival data. *Statistics in Medicine*, 36(30):4831–4842, 2017.
- [127] F. Gao, D. Zeng, and D. Y. Lin. Semiparametric estimation of the accelerated failure time model with partly interval-censored data. *Biometrics*, 73(4):1161–1168, 2017.
- [128] S. Han, A. C. Andrei, and K. W. Tsui. A semiparametric regression method for interval-censored data. *Communications in Statistics - Simulation and Computation*, 43(1):18–30, 2014.
- [129] M. H. Dehghan and T. Duchesne. On the performance of some non-parametric estimators of the conditional survival function with interval-censored data. *Computational Statistics & Data Analysis*, 55(12):3355–3364, 2011.
- [130] N. Pantazis, M. G. Kenward, and G. Touloumi. Performance of parametric survival models under non-random interval censoring: a simulation study. *Computational Statistics & Data Analysis*, 63:16–30, 2013.
- [131] J. M. Williamson, G. A. Satten, J. A. Hanson, H. Weinstock, and S. Datta. Analysis of dynamic cohort data. *American Journal of Epidemiology*, 154(4):366–372, 2001.
- [132] M. J. Crowther and P. C. Lambert. Simulating biologically plausible complex survival data. *Statistics in Medicine*, 32(23):4118–4134, 2013.
- [133] I. R. White. simsum: analyses of simulation studies including Monte Carlo error. *The Stata Journal*, 10(3):369–385, 2010.
- [134] C. Anderson-Bergman. icenReg: Regression models for interval censored data in R. *Journal of Statistical Software*, 81(12):1–23, 2017.
- [135] T. M. Therneau, T. Lumley, E. Atkinson, and C. Crowson. *survival: survival analysis*. R package, version 3.3-1. Available from CRAN at <https://CRAN.R-project.org/package=survival>.
- [136] S. Dryden-Peterson, M. Bvochora-Nsingi, G. Suneja, J. A. Efstatthiou, S. Grover, S. Chiyapo, et al. HIV infection and survival among women with cervical cancer. *Journal of Clinical Oncology*, 34(31):3749–3757, 2016.
- [137] J. W. Lewin, N. A. O'Rourke, A. K. H. Chiow, R. Bryant, I. Martin, L. K. Nathanson, et al. Long-term survival in laparoscopic vs open resection for colorectal liver metastases: inverse probability of treatment weighting using propensity scores. *HPB*, 18(2):183–191, 2016.

- [138] D. Westreich, S. R. Cole, E. F. Schisterman, and R. W. Platt. A simulation study of finite-sample properties of marginal structural Cox proportional hazards models. *Statistics in Medicine*, 31(19):2098–2109, 2012.
- [139] Y. Xiao, M. Abrahamowicz, and E. E. M. Moodie. Accuracy of conventional and marginal structural Cox model estimators: a simulation study. *The International Journal of Biostatistics*, 6(2):Article 13, 2010.
- [140] O. O. Aalen, R. J. Cook, and K. Røysland. Does Cox analysis of a randomized survival study yield a causal treatment effect? *Lifetime Data Analysis*, 21(4):579–593, 2015.
- [141] H. Uno, B. Claggett, L. Tian, E. Inoue, P. Gallo, T. Miyata, et al. Moving beyond the hazard ratio in quantifying the between-group difference in survival analysis. *Journal of Clinical Oncology*, 32(22):2380–2385, 2014.
- [142] H. Uno, J. Wittes, H. Fu, S. D. Solomon, B. Claggett, L. Tian, et al. Alternatives to hazard ratios for comparing the efficacy or safety of therapies in noninferiority studies. *Annals of Internal Medicine*, 163(2):127–134, 2015.
- [143] S. R. Cole and M. A. Hernán. Adjusted survival curves with inverse probability weights. *Computer Methods and Programs in Biomedicine*, 75(1):45–49, 2004.
- [144] J. Xie and C. Liu. Adjusted Kaplan-Meier estimator and log-rank test with inverse probability of treatment weighting for survival data. *Statistics in Medicine*, 24(20):3089–3110, 2005.
- [145] S. C. Conner, L. M. Sullivan, E. J. Benjamin, M. P. LaValley, S. Galea, and L. Trinquart. Adjusted restricted mean survival times in observational studies. *Statistics in Medicine*, 38(20):3832–3860, 2019.
- [146] S. L. T. Normand, M. B. Landrum, E. Guadagnoli, J. Z. Ayanian, T. J. Ryan, P. D. Cleary, et al. Validating recommendations for coronary angiography following acute myocardial infarction in the elderly: a matched analysis using propensity scores. *Journal of Clinical Epidemiology*, 54(4):387–398, 2001.
- [147] P. C. Austin. Balance diagnostics for comparing the distribution of baseline covariates between treatment groups in propensity-score matched samples. *Statistics in Medicine*, 28(25):3083–3107, 2009.
- [148] D. Hajage, F. Tubach, P. G. Steg, D. L. Bhatt, and Y. De Rycke. On the use of propensity scores in case of rare exposure. *BMC Medical Research Methodology*, 16:38, 2016.
- [149] W. G. Havercroft and V. Didelez. Simulating from marginal structural models with time-dependent confounding. *Statistics in Medicine*, 31(30):4190–4206, 2012.
- [150] A. Cronin, L. Tian, and H. Uno. strmst2 and strmst2pw: new commands to compare survival curves using the restricted mean survival time. *The Stata Journal*, 16(3):702–716, 2016.

- [151] A. Chatton, F. Le Borgne, C. Leyrat, and Y. Foucher. G-computation and doubly robust standardisation for continuous-time data: a comparison with inverse probability weighting. *Statistical Methods in Medical Research*, 31(4):706–718, 2022.
- [152] R. H. Keogh, S. R. Seaman, J. M. Gran, and S. Vansteelandt. Simulating longitudinal data from marginal structural models using the additive hazard model. *Biometrical journal*, 63(7):1526–1541, 2021.
- [153] M. Á. Hernán, B. Brumback, and J. M. Robins. Marginal structural models to estimate the causal effect of zidovudine on the survival of HIV-positive men. *Epidemiology*, 11(5):561–570, 2000.
- [154] J. K. Lunceford and M. Davidian. Stratification and weighting via the propensity score in estimation of causal treatment effects: a comparative study. *Statistics in Medicine*, 23(19):2937–2960, 2004.
- [155] P. J. Huber. The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the 5th Berkeley symposium on mathematical statistics and probability*, volume 1, pages 221–233. University of California Press, 1967.
- [156] H. White. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica*, 48(4):817–838, 1980.
- [157] H. White. Maximum likelihood estimation of misspecified models. *Econometrica*, 50(1):1–25, 1982.
- [158] D. Y. Lin and L. J. Wei. The robust inference for the Cox proportional hazards model. *Journal of the American Statistical Association*, 84(408):1074–1078, 1989.
- [159] J. C. Deville. Variance estimation for complex statistics and estimators: linearization and residual techniques. *Survey Methodology*, 25(2):193–204, 1999.
- [160] D. A. Binder. Fitting Cox’s proportional hazards models from survey data. *Biometrika*, 79(1):139–147, 1992.
- [161] T. Cai, R. J. Hyndman, and M. P. Wand. Mixed model-based hazard estimation. *Journal of Computational and Graphical Statistics*, 11(4):784–798, 2002.
- [162] L. A. Stefanski and D. D. Boos. The calculus of M-estimation. *The American Statistician*, 56(1):29–38, 2002.
- [163] W. W. Loh and S. Vansteelandt. Confounder selection strategies targeting stable treatment effect estimators. *Statistics in Medicine*, 40(3):607–630, 2021.
- [164] B. C. Saul and M. G. Hudgens. The calculus of M-estimation in R with geex. *Journal of Statistical Software*, 92(2):1–15, 2020.

- [165] P. C. Lambert. *standsurv: Stata module to compute standardized (marginal) survival and related functions*, 2021. Available from the SSC archive. See also <https://econpapers.repec.org/software/bocbocode/s458991.htm>.
- [166] F. Ieva, C. H. Jackson, and L. D. Sharples. Multi-state modelling of repeated hospitalisation and death in patients with heart failure: the use of large administrative databases in clinical epidemiology. *Statistical Methods in Medical Research*, 26(3):1350–1372, 2017.
- [167] G. Manzini, T. J. Ettrich, M. Kremer, M. Kornmann, D. Henne-Brunns, D. A. Eikema, et al. Advantages of a multi-state approach in surgical research: how intermediate events and risk factor profile affect the prognosis of a patient with locally advanced rectal cancer. *BMC Medical Research Methodology*, 18: 23, 2018.
- [168] S. Gilard-Pioc, M. Abrahamowicz, A. Mahboubi, A. M. Bouvier, O. Dejardin, E. Huszti, et al. Multi-state relative survival modelling of colorectal cancer progression and mortality. *Cancer Epidemiology*, 39(3):447–455, 2015.
- [169] C. Eulenburg, S. Mahner, L. Woelber, and K. Wegscheider. A systematic model specification procedure for an illness-death model without recovery. *PLoS ONE*, 10(4):e0123489, 2015.
- [170] J. G. Le-Rademacher, R. A. Peterson, T. M. Therneau, B. L. Sanford, R. M. Stone, and S. J. Mandrekar. Application of multi-state models in cancer clinical trials. *Clinical Trials*, 15(5):489–498, 2018.
- [171] M. Nazari, S. H. Nazari, F. Zayeri, M. G. Dehaki, and A. A. Baghban. Estimating transition probability of different states of type 2 diabetes and its associated factors using Markov model. *Primary Care Diabetes*, 12(3):245–253, 2018.
- [172] L. C. de Wreede, M. Fiocco, and H. Putter. mstate: an R package for the analysis of competing risks and multi-state models. *Journal of Statistical Software*, 38(7):1–30, 2011.
- [173] C. H. Jackson. flexsurv: a platform for parametric survival modeling in R. *Journal of Statistical Software*, 70(8):1–33, 2016.
- [174] M. J. Crowther. merlin - a unified modelling framework for data analysis and methods development in Stata. *The Stata Journal*, 20(4):763–784, 2020.
- [175] M. J. Crowther. Extended multivariate generalised linear and non-linear mixed effects models. *arXiv preprint*, arXiv:1710.02223, 2017. Preprint at <https://arxiv.org/abs/1710.02223>.
- [176] M. Schumacher, M. Wangler, M. Wolkewitz, and J. Beyersmann. Attributable mortality due to nosocomial infections. A simple and useful application of multistate models. *Methods of Information in Medicine*, 46(5):595–600, 2007.

- [177] R. Bender, T. Augustin, and M. Blettner. Generating survival times to simulate Cox proportional hazards models. *Statistics in Medicine*, 24(11):1713–1723, 2005.
- [178] S. K. Metzger and B. T. Jones. mstatecox: a package for simulating transition probabilities from semiparametric multistate survival models. *The Stata Journal*, 18(3):533–563, 2018.
- [179] R. J. Cook and J. F. Lawless. Statistical issues in modeling chronic disease in cohort studies. *Statistics in Biosciences*, 6:127–161, 2014.
- [180] C. Jackson, J. Stevens, S. Ren, N. Latimer, L. Bojke, A. Manca, et al. Extrapolating survival from randomized trials using external data: a review of methods. *Medical Decision Making*, 37(4):377–390, 2017.
- [181] National Institute for Health and Care Excellence. Guide to the methods of technology appraisal. Available from <https://www.nice.org.uk/article/pmg9>.
- [182] A. Sjölander. Regression standardization with the R package stdReg. *European Journal of Epidemiology*, 31(6):563–574, 2016.
- [183] J. M. Gran, S. A. Lie, I. Øyeften, Ø. Borgan, and O. O. Aalen. Causal inference in multi-state models—sickness absence and work for 1145 participants after work rehabilitation. *BMC Public Health*, 15:1082, 2015.
- [184] A. van den Hout. *Multi-state survival models for interval-censored data*. Chapman and Hall/CRC, 2017.
- [185] D. Commenges. Inference for multi-state models from interval-censored data. *Statistical Methods in Medical Research*, 11(2):167–182, 2002.
- [186] F. Gillaizeau, T. Sénage, F. Le Borgne, T. Le Tourneau, J. C. Roussel, K. Lefondrè, et al. Inverse probability weighting to control confounding in an illness-death model for interval-censored data. *Statistics in Medicine*, 37(8):1245–1258, 2018.
- [187] A. Gasparini, T. P. Morris, and M. J. Crowther. INTEREST: INteractive Tool for Exploring REsults from Simulation sTudies. *Journal of Data Science, Statistics, and Visualisation*, 1(4):1–22, 2021.