Recurrent Episode Analyses of Pulmonary Exacerbations in the Infant Study of Inhaled Saline (ISIS) Trial

Arthur C. Baines

A thesis

submitted in partial fulfillment of the

requirements for the degree of

Master of Science

University of Washington

2013

Committee:

Lyndia Brumback

Susanne May

Program Authorized to Offer Degree:

Biostatistics

University of Washington

Abstract

Recurrent Episode Analyses of Pulmonary Exacerbations

in the Infant Study of Inhaled Saline (ISIS) Trial

Arthur C. Baines

Chair of the Supervisory Committee:

Research Associate Professor Lyndia Brumback

Biostatistics

The Infant Study of Inhaled Saline (ISIS) randomized, controlled clinical trial evaluated the use of daily inhaled hypertonic saline treatment for 48 weeks to reduce the rate of pulmonary exacerbations in children of less than 6 years of age with cystic fibrosis. In an earlier trial among older children and adults with cystic fibrosis, the group randomized to inhaled hypertonic saline had significantly fewer pulmonary exacerbations and a significantly higher proportion of participants without exacerbations. In ISIS, the primary analysis of rate of pulmonary exacerbations using Poisson regression had a null result for treatment effect (the rates were 2.3 per person-year for both groups; adjusted rate ratio = 0.98 with a 95% confidence interval of

0.84, 1.15), as did an additional analysis of exacerbation-free survival using Cox proportional hazards regression. However, these analyses have limitations. The primary analysis fails to account for the timing of exacerbations, while the second analysis ignores all but the first exacerbation. This thesis applies extensions of the Cox proportional hazards model and a relatively new method, temporal process regression, to provide a more comprehensive picture of the ISIS trial pulmonary exacerbation data in an exploratory setting. Conclusions regarding the effect of treatment from these additional analyses are consistent with those of the original ISIS analyses. An example dataset is constructed to illustrate how extensions of the proportional hazards model and temporal process regression may reveal evidence of a treatment effect that goes undetected by the Poisson regression and simple Cox model analyses.

Table of Contents

1. **Introduction**

This thesis focuses on analyzing data from clinical trials with discrete outcomes where the timing of those outcomes is of interest (commonly referred to as time-to-event data). In particular, we examine data from the Infant Study of Inhaled Saline (ISIS) clinical trial. The ISIS trial investigated an inhaled saline treatment for infants and young children with cystic fibrosis, where the discrete outcome was pulmonary exacerbations. These exacerbations are characterized by a temporary decline in lung function due to inflammation and/or infection. They are a common endpoint of interest for trials in cystic fibrosis and chronic obstructive pulmonary disease.

Survival analysis was initially concerned with data where the time to a single outcome is the primary interest (for example, time until death). In the ISIS trial the pulmonary exacerbation outcome can occur more than once for each individual. This data structure can be found in a number of disease settings and data of this type are known as recurrent event data. An earlier example of recurrent event data within the cystic fibrosis field is the data from the rhDNase trial, which evaluated the effects of twice-daily administration of dornase alfa on the risk of pulmonary exacerbations. [1] Methodological research has been done to extend the most commonly used technique in survival analysis, the Cox proportional hazards regression model, to accommodate recurrent event data.

With the Cox proportional hazards model and its extensions, the primary model component is the instantaneous event rate over time, or the hazard function. Another approach to modeling recurrent event data is through modeling the expected number of events per individual by a given time, referred to as a mean model. In general for recurrent event data, these two families of

models are reformulations of one another. [2] However, when the events in question have durations associated with them, as they do with pulmonary exacerbations, the relationship between families does not hold. [2]

In this thesis, we examine the application of extensions of the Cox model to the recurrent event data from the ISIS clinical trial. Additionally, we apply a newer variant of a mean model, called temporal process regression, to the ISIS data. Finally, we construct an example dataset to investigate characteristics of recurrent event data that can be detected by these newer methods but are not seen by more traditional models.

This thesis is organized as follows. In Section 2, we provide an introduction to cystic fibrosis and the ISIS clinical trial. In Section 3, we discuss the existing methods for recurrent event data based on extending the Cox proportional hazards model as well as the newer temporal process regression method. In Sections 4 and 5, we apply these methodologies to the ISIS trial data. In Section 6, we construct an example dataset to illustrate that the extensions of the proportional hazards model and temporal process regression may reveal evidence of a treatment effect that goes undetected by Poisson regression and simple Cox model analyses. We also provide simulations to demonstrate the statistical properties of some of the extensions of the proportional hazards model. In Section 7, we provide conclusions and potential directions for future work in this area. In the appendices, we provide details about the definition of a pulmonary exacerbation and code to implement the methods described throughout this thesis.

2. The ISIS Clinical Trial

2.1. Cystic Fibrosis

Cystic fibrosis (CF) is a genetic lung disease characterized by chronic airway inflammation, infection and obstruction. Onset of CF associated symptoms, such as increased mucus production, persistent cough, shortness of breath, poor growth and poor weight gain typically begins at infancy and these symptoms endure throughout life. There is currently no cure for CF. Inhaled, oral, and intravenous antibiotics are used quite aggressively to treat individuals with CF, usually with the intention of suppressing lung and airway infections. Other treatments are regularly used to facilitate clearance of mucosal buildup in airways to prevent lung damage. According to the Cystic Fibrosis Foundation, roughly 1,000 new cases of CF are diagnosed each year and the current predicted median age of survival for a person with CF is in the late 30s.

Notable clinical trials in CF in the past 15 years include: a 24 week trial which reported that intermittent administration of inhaled tobramycin improved pulmonary function, [3] another 24 week trial which reported that children and adolescents with CF uninfected with *P. aeruginosa* treated with azithromycin did not experience improved pulmonary function, [4] and a 2006 study by Elkins, et al. which reported hypertonic saline as being effective in reducing the mean number of pulmonary exacerbations per participant based on a clinical trial among children older than 6 years and adults with CF. [5]

2.2. ISIS Trial Design

The ISIS clinical trial is a multicenter, randomized, double-blind, controlled clinical trial evaluating the effectiveness of inhaled hypertonic saline as a treatment for young children (less

than six years of age) and infants with cystic fibrosis (CF). The primary outcome was the rate of protocol defined pulmonary exacerbations per person-year as observed over the 48-week study (see Appendix A for the ISIS protocol definition of a pulmonary exacerbation). [6]

Participants who were between the ages of 4 months and 2 years at enrollment and from a predetermined subset of the study centers were asked to participate in an infant pulmonary function testing (iPFT) sub-study. As a part of the iPFT sub-study, baseline lung function tests were administered to obtain a number of measures of lung function, which is considered an important risk factor for pulmonary exacerbations.

2.3. **ISIS Results**

The ISIS trial was conducted between April 2009 and October 2011 with 158 participants randomized to hypertonic saline and 163 to isotonic saline (control). Of the 321 randomized participants, 73 were also enrolled into the iPFT sub-study.

The primary endpoint of the ISIS trial did not provide evidence in favor of a treatment effect: the adjusted Poisson regression model of pulmonary exacerbation counts on treatment arm resulted in an estimated rate ratio comparing individuals on hypertonic saline to those on isotonic saline of 0.98 with a 95% C.I of (0.84, 1.15). The rates were 2.3 exacerbations per person-year of observation time for both groups.

Additionally, both a Cox proportional hazards model of time to first pulmonary exacerbation and a linear regression model of the log of antibiotic treatment days (for pulmonary exacerbations)

had null results, with a hazard ratio of 0.94, 95% C.I of (0.74, 1.21) and a ratio of mean total number of antibiotic treatment days of 1.13, 95% C.I. of (0.91, 1.40) comparing hypertonic saline to isotonic saline, respectively. [6]

3. Extensions of the Cox Proportional Hazards Model for Recurrent Events/Episodes

While the analyses reported in the primary manuscript of the ISIS trial by Rosenfeld et al. did not yield evidence of a hypertonic saline treatment effect, those analyses failed to take into account a portion of the pulmonary exacerbation information we have from the ISIS trial data. In order to conduct the Poisson rate regression, it was necessary to collect information on the number of exacerbations experienced. By contrast, the Cox proportional hazards regression required accurate information about times until first exacerbations. In addition to those key components, we have data that was gathered to assess the total number of antibiotic treatment days for exacerbations. In other words, we have the start and stop times of antibiotic courses associated with each exacerbation, and thus, an estimate of the duration of those exacerbations. This means the ISIS data provide a record of the time on study for each subject, including approximate start and stop times for each event (pulmonary exacerbation) of every subject over the duration of the study. Data of this type are referred to as recurrent event data. [7,8,9] More recently, data of this type have been labeled recurrent episode data if the events have durations. [2] Although the analyses of Rosenfeld et al. did account for all pulmonary exacerbations experienced in their Poisson regression model and did model the times to first exacerbations, they were unable to incorporate the timing of any exacerbation beyond a participant's first event into their analyses. The primary reason for this shortcoming is that the Cox model and the Poisson regression method used in their analyses are insufficient to fully model recurrent event or recurrent episode

data. However, a number of viable extensions of the Cox proportional hazards model exist to model recurrent event data and a method has recently been proposed to analyze recurrent episode data. Each extension of the Cox model makes a different set of assumptions to accommodate recurrent event data, and in most cases these differences are not trivial. An introduction and discussion of these differences follows.

3.1. **Andersen-Gill**

The most straight-forward of the recurrent event methods is known as the Andersen-Gill (AG) model. [10] This model makes two main assumptions: the recurrent events are independent within each subject, and all events are considered to be of the same type. In other words, any ordering or differentiation in event type cannot be modeled with this method. The obvious correlation between a single individual's events can be addressed to some extent by allowing for intra-subject correlation (clustering on subject).

3.2. **Marginal (Wei, Lin and Weissfeld)**

A second method for modeling recurrent event data is the marginal model proposed by Wei, Lin and Weissfeld. [11] The key assumption of the marginal model is that an individual is at risk for all events if they are at risk for any. This can be useful in situations where the events being considered are of different types and are not dependent on one another (i.e. an event of type A is not required for a subject to have an event of type B and an event of type A does not rule out an event of type B), but becomes an issue when events are inherently ordered. For example, under the marginal model, an individual would be modeled as at risk for their first and second events

concurrently, even though in reality they must experience the first before the second. Most settings where there is a single event (or episode) type are inherently ordered.

3.3. **Conditional (Prentice, Williams and Peterson)**

The conditional model of Prentice, Williams and Peterson is the third method we will consider. [12] The primary assumption of this model is that it assumes a subject is not at risk for an event until they have experienced all preceding events. Namely, in a setting with ordered events, an individual would not be considered at risk for their third event until both their first and second had occurred. The implication here is that a subject would have (perhaps considerably) less time at risk for their third event under this model than they would under the marginal model introduced above. A caveat of the conditional model is that all inferences are made conditionally, resulting in more specialized interpretations.

3.4. **Implementation**

These three extensions of the Cox proportional hazards regression model can be implemented using standard statistical software, such as R or Stata. The most difficult aspect of this implementation is the data management necessary to construct the required datasets. Therneau and Grambsch demonstrate the application of the AG, marginal, and conditional models to data from the rhDNase clinical trial using R. [8] Like the ISIS data, the rhDNase data also originate from a trial investigating a treatment for individuals with cystic fibrosis where both the time to first pulmonary exacerbation and times to subsequent exacerbations were recorded. Notes on verification of their results as well as the Stata code used are included in Appendix B (Sections 10.1 and 10.3, respectively).

3.5. **Discussion of Cox Methods and ISIS data**

These three models all have their advantages and disadvantages in the setting of the ISIS clinical trial with recurrent events in the form of pulmonary exacerbations. The Andersen-Gill model, while easy to implement and interpret, is perhaps too restrictive in what it allows. The marginal model has fundamental issues with ordered events, which we need to be able to address here. Finally, the conditional model has issues with regard to ease of interpretability, but it appears to make the most intuitive sense of the three for the ISIS data.

A common characteristic of the preceding methods is that they do not model event durations when applied to recurrent episode data. With the ISIS data these durations are of interest in and of themselves, and we would like to explicitly model them in addition to the number of exacerbations.

4. **Temporal Process Regression**

Temporal process regression (TPR), first proposed by Fine, Yan and Kosorok in 2004, [13] is a method for modeling recurrent episode data. A follow-up publication in 2008 by Yan and Fine details the application of their method to the rhDNase data that was used by Therneau and Grambsch. [2] TPR is a form of generalized linear model (GLM) that allows for the modeling of so-called temporal processes, such as the total number of events or episodes over time or the number of accumulated days in episodes over time within a mean model framework. TPR models produce non-parametric, time-varying coefficient estimates for the effect of fixed (not time-varying) covariates. By modeling both the total number of exacerbations over time and the

days in exacerbation over time from the ISIS data using TPR, we can get a more comprehensive picture of recurrent exacerbation behavior over the duration of the study that includes explicit consideration for the duration of exacerbations.

Let t denote time, X a covariate vector, $Y(t)$ a response vector, and $\delta(t) = I(t < C)$ a data availability indicator, where C is the follow up time. Then given a predetermined time window $[l, u]$ within which we observe n independent and identically distributed copies of $\{Y(t), X: \delta(t)\}$ denoted $\{Y_i(t), X_i: \delta_i(t) = 1\}$ for $i = 1, \ldots, n$ where $\delta_i(t) = 1$ for all $t < C_i$, the temporal process regression GLM is:

$$h\{\mu_i(t)\} = X_i^T \beta(t),$$

where $\mu_i(t) = E\{Y_i(t)|X_i\}$, h is a known link function, and $\beta(t)$ is a vector of completely unspecified time-varying coefficients.[2] $Y_i(t)$ could be the total number of events by time t for participant i, or the total duration of episodes by time t for participant i, for example. The estimation of $\beta(t)$ is achieved through the use of a "snapshot" cross-sectional dataset at each time $t \in [l, u]$. The parameters of the GLM at time t are then estimated by solving the quasi-score equation:

$$\sum_{i=1}^{n} U_i\{\beta(t)\} = 0,$$

where $U_i\{\beta(t)\} = \delta_i(t) D_i\{\beta(t)\} V_i^{-1}\{\beta(t)\}\{Y_i(t) - \mu_i(t)\}$, $D_i\{\beta(t)\} = d\mu_i(t)/d\beta(t)$, and $V_i^{-1}\{\beta(t)\}$ is a working weight function. The combination of $\hat{\beta}(t)$ from the quasi-score equation for all time points in $[l, u]$ gives the non-parametric time-varying estimator of $\beta(t)$.

Yan and Fine also present methods for hypothesis testing. For example, they suggest a naïve (and straight forward) approach of testing at a few specified time points using a quadratic form statistic. Ultimately, they recommend the use of an integral test statistic that is applied across all of $[l, u]$. Given the null hypothesis $H_0 : \beta_i(t) = c(t)$ for $t \in [l, u]$, where $c(t)$ is a known function of t, the integral test statistic takes the form:

$$T = \hat{\Sigma}_{\Delta}^{-1/2}\Delta,$$

where $\Delta = \int_l^u \{\hat{\beta}(t) - c(t)\}\widetilde{W}(t)dt$, $\hat{\Sigma}_{\Delta}$ is the estimated variance of Δ, and $\widetilde{W}(t)$ is a non-negative weight function. One disadvantage of this test statistic is the potential for deviations from the null in opposite directions to cancel out in the integrand.

4.1. **Implementation of TPR**

The R package '`tpr`' contains functions for fitting temporal process regression models as well as tools for evaluating those models and some example code utilizing the data from the rhDNase trial. [14] Verification of the time-varying coefficient plots in Figure 4 of Yan and Fine (2008) was successful after a number of modifications were made to the example code and dataset. Despite these changes, the integral test statistics presented in their paper (Table 2) could not be verified (before or after the above modifications were made). Dr. Yan was contacted in April of 2013 and was also unsure of the reason for this discrepancy. As a result, the TPR integral test statistics are not presented for our analyses. Furthermore, visual plots of the coefficients over time provide a better description of the TPR models for our exploratory purposes. See Appendix B for more details regarding this verification process and for the R code used (Sections 10.2 and 10.4, respectively).

5. **Method Application to ISIS Data**

Before discussing the application of the extensions of the Cox model and TPR to the ISIS data, it is important to note that all analyses conducted here are exploratory in nature. They are intended to further our understanding of the data from the ISIS trial as well as explore what these more advanced models might detect that the Poisson regression and Cox proportional hazards models could not. As such, no corrections for multiple comparisons are made and nominal 95% confidence intervals are presented in all scenarios. Hazard ratios and confidence intervals are presented instead of unexponentiated coefficients and standard errors to aid in interpreting and understanding model fits.

5.1. **All Randomized Participants**

We begin our analysis of the ISIS data by looking at the entire study population. Table 1 contains descriptive statistics of baseline characteristics by arm for the 321 participants; characteristics were similar in the two arms. See Rosenfeld et al. for additional baseline information and a consort diagram for the ISIS study. [6]

Table 1: Baseline characteristics by treatment arm for ISIS participants

Characteristic	Hypertonic Saline (n = 158)	Isotonic Saline (n = 163)
Age, mean (SD) (years)	2.2 (1.4)	2.3 (1.5)
Male, No. (%)	84 (53)	92 (56)
Weight, mean (SD), kg	12.2 (4.1)	12.5 (4.1)
Height, mean (SD), cm	84.8 (14.8)	85.7 (15)

Participants had between zero and nine exacerbations over the study duration with the majority having between zero and three. Table 2 shows the frequency of exacerbation counts by treatment arm.

Table 2: Frequencies of exacerbations by treatment arm

Exacerbation Count	0	1	2	3	4	5	6	7	8	9	Total
Isotonic Saline	34	39	35	25	15	6	4	2	1	2	163
Hypertonic Saline	34	33	36	29	9	11	4	2	0	0	158
Total	68	72	70	54	24	17	8	4	1	2	321

When making inferences about pulmonary exacerbations beyond the first exacerbation, we must decide what to do with the information we have on the risk of exacerbations beyond the third or fourth, where the data become sparse. Only 15/163 (9%) of the participants in the isotonic saline arm and 17/158 (11%) of the participants in the hypertonic saline arm had more than four exacerbations (18% and 16% had more than three exacerbations, respectively). As Therneau and Grambsch noted, there are at least three options. [8] The first option is to use all of the information as it is and be aware that any estimates for the risk of exacerbations at the higher exacerbation counts, or strata, will probably be highly variable and/or unstable; the second option is to truncate the data and discard the information beyond a certain strata, and the third is to collapse the information from the highest counts down into some cutoff strata that is reasonable. The modeling implication for the AG, marginal, and conditional models of the last option is that the risk of exacerbations at or beyond the cutoff strata is assumed to be the same. Which cutoff strata is used could be a scientific argument (e.g. it is argued that the risk of an event beyond a certain count is approximately the same), or it could be a decision based on how sparse data become in higher strata, as appears to be the case for Therneau and Grambsch.[8]

Moving forward we will examine using the information as it is and collapsing the information into lower strata when looking at an overall treatment effect, and collapsing the information into lower strata when considering an interaction between the treatment effect and the exacerbation

strata. For the collapsed strata analyses, cutoffs of the third and fifth exacerbation will be explored. For example, when using the third exacerbation as a cutoff, our inferences are limited to effects regarding the first exacerbation, second exacerbation and exacerbations beyond the second. All models presented using data from all randomized participants are fit with treatment as the only covariate; additional covariates will be introduced when we consider the iPFT sub-study data.

5.1.1. **Extensions of the Cox Proportional Hazards Model**

As previously stated, analysis of time to first pulmonary exacerbation using a Cox proportional hazards model resulted in an estimated hazard ratio of 0.94, comparing individuals on hypertonic saline (HS) to those on the control arm, isotonic saline (IS) (95% C.I. of (0.74, 1.21)). The accompanying Kaplan-Meier survival estimates are plotted in Figure 1.

The AG model, which models all pulmonary exacerbations over the course of the study, results in an estimated hazard ratio of 0.98, comparing individuals on HS to those on IS (95% C.I. of (0.80, 1.21)). One interpretation of this hazard ratio is that individuals who are on HS are estimated to experience exacerbations at a 2% lower rate than those on IS (Table 3). When compared to the hazard ratio of 0.94 from the Cox model of time to first exacerbation, this suggests that exacerbations beyond the first occurred more frequently in the HS arm. In other words, there is also a lack evidence of a treatment effect when considering exacerbations beyond the first using the AG model.

The marginal model also accounts for exacerbations beyond the first, but allows for different baseline hazard functions by exacerbation strata and considers participants at risk for all

exacerbations upon enrollment. The model fit results in an estimated hazard ratio of 0.91, comparing individuals on HS to those on IS (95% C. I. of (0.68, 1.21)). That is, individuals on HS are estimated to experience exacerbations at a 9% lower rate than those on IS under this model's assumptions. This estimate is unchanged based on the number of exacerbation strata considered (Table 3). When compared to the hazard ratio from the Cox proportional hazards model of time to first exacerbation, this model suggests that exacerbations beyond the first occur less frequently in the HS study arm. While this result is counter to that of the AG model, the confidence intervals are similar and the marginal model's assumptions are ill suited to the recurrent exacerbations setting (see Section 3.2).

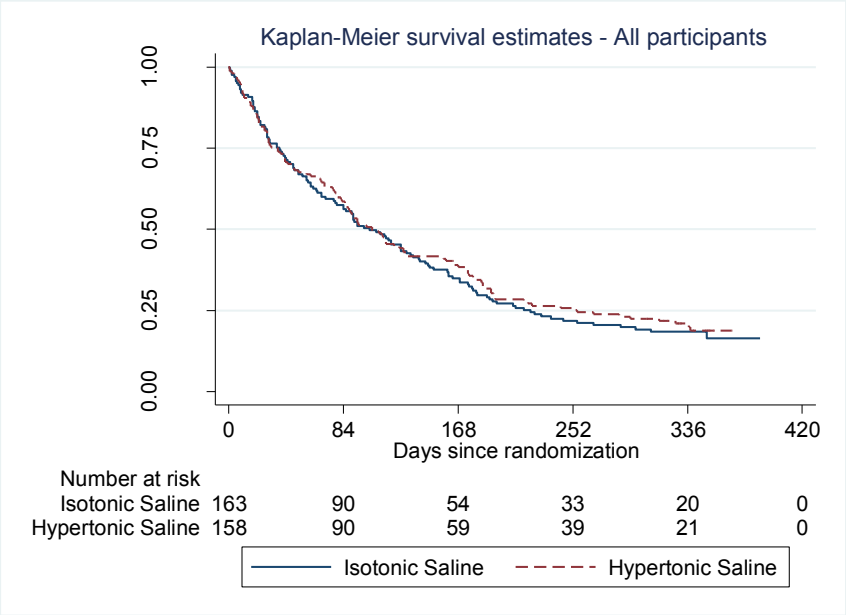

Figure 1: K-M survival estimates by treatment arm - time to first exacerbation

The conditional model also considers exacerbations beyond the first, while allowing for different baseline hazards by exacerbation strata. However, it views participants as being at risk for the j^{th} exacerbation only after they have had the previous j-1 exacerbations. The model fit results in an estimated hazard ratio of 1.00 comparing individuals on HS to those on IS (95% C. I. of (0.86,

1.18)). That is, individuals on HS are estimated to experience exacerbations at the same rate as those on IS under this model's assumptions. This estimate is virtually unchanged based on the number of exacerbation strata considered (Table 3). When compared to the hazard ratio from the Cox proportional hazards model of time to first exacerbation, this model suggests that exacerbations beyond the first occurred more frequently in the HS study arm, as was seen with the AG model.

Table 3: Cox model treatment effect estimates

Model	Treatment HR (95% C.I.)
Time to First Exacerbation	0.94 (0.74, 1.21)
Andersen-Gill	0.98 (0.79, 1.21)
Conditional	1.00 (0.86, 1.17)
Conditional – 3 Strata	0.99 (0.84, 1.17)
Conditional – 5 Strata	0.99 (0.85, 1.16)
Marginal	0.91 (0.68, 1.21)
Marginal – 3 Strata	0.91 (0.69, 1.20)
Marginal - 5 Strata	0.91 (0.68, 1.21)

Both the marginal and conditional models allow consideration of an interaction effect between treatment and exacerbation strata. In other words, the models allow the treatment effect to be different for the different events. This is where these two models further differentiate themselves from the Andersen-Gill model. We consider these interaction models using the two different collapsed exacerbation strata scenarios, the first with three strata and the second with five. With three strata the assumption is that the treatment effect might be different for the first and second exacerbation but it is assumed to be the same for all exacerbations beyond the second; the situation is analogous with five strata.

The marginal model with the three strata treatment interaction results in estimated hazard ratios of 0.94 for the first event, 1.04 for the second and 0.77 for events beyond the second, with all 95% confidence intervals including the null value of one (Table 4). The point estimates indicate that the hazard for second exacerbations was observed to be higher in the HS arm compared to the IS arm while the hazard for exacerbations beyond the second was seen to be higher in the IS arm. However, we expect some variation in these estimates by chance and the corresponding confidence intervals do not exclude one.

The conditional model with the three strata treatment interaction results in estimated hazard ratios of 0.94 for the first event, 1.13 for the second (conditional on having had a first event) and 0.94 for events beyond the second (conditional on having had a first and second event), with all 95% confidence intervals including one (Table 4).

Table 4: Marginal and conditional models - 3 strata treatment effect

Marginal	Treatment HR (95% C.I)	Conditional	Treatment HR (95% C.I)
First	0.94 (0.74, 1.21)	**First**	0.94 (0.74, 1.21)
Second	1.04 (0.77, 1.40)	**Second**	1.13 (0.84, 1.51)
Third+	0.77 (0.49, 1.22)	**Third+**	0.94 (0.69, 1.28)

The marginal and conditional models with five strata treatment interactions are summarized in Table 5. The most apparent difference between these two models occurs for exacerbations beyond the fourth. For those exacerbations, the conditional model suggests an elevated (but not significantly different) risk for the HS arm compared to the IS control arm (similar to its estimates for second and third exacerbations), while the marginal model estimates a hazard ratio favoring the HS arm (also not statistically significant, however). This difference is most likely due to how participants are considered at risk for exacerbations beyond the fourth (as well as

prior exacerbations). To be explicit, the conditional model only considers those who have had a fourth exacerbation and the marginal considers everyone who is not currently in an exacerbation episode at risk for a fifth or higher exacerbation, even if they had not had a forth, third, second, or first exacerbation. For this reason, the estimates provided by the conditional model seem much more reasonable.

The differences in estimates of treatment effect for exacerbations beyond the third between the three strata models (Table 4) and the five strata models (Table 5) suggest that assuming the treatment effect is the same for exacerbations beyond the third may be inappropriate, even if the information gained about higher exacerbation strata is limited. The biology behind such a possible differential treatment effect is not clear, and these observed differences may be due to random variation.

Table 5: Marginal and conditional models - 5 strata treatment effect

Marginal	Treatment HR (95% C.I.)	Conditional	Treatment HR (95% C.I.)
First	0.94 (0.74, 1.21)	**First**	0.94 (0.74, 1.21)
Second	1.04 (0.77, 1.40)	**Second**	1.13 (0.84, 1.51)
Third	1.04 (0.71, 1.52)	**Third**	1.14 (0.79, 1.66)
Fourth	0.86 (0.50, 1.46)	**Fourth**	0.58 (0.32, 1.07)
Fifth+	0.29 (0.08, 1.01)	**Fifth+**	1.08 (0.63, 1.85)

5.1.2. **Temporal Process Regression**

We begin our application of TPR using data from all 321 ISIS participants, as we did with the extensions of the Cox model. We examine the total number of exacerbations over time process model as well as the accumulated days in exacerbation over time process model to account for exacerbation duration. These models are fit with treatment as the only covariate.

The total number of exacerbations process model was fit using a log link. Plots of the exponentiated coefficient estimates and pointwise 95% confidence intervals for the mean number of exacerbations over time in the IS arm (the model intercept) and the ratio of the mean number of exacerbations over time comparing HS to IS (treatment effect) are presented in Figure 2.

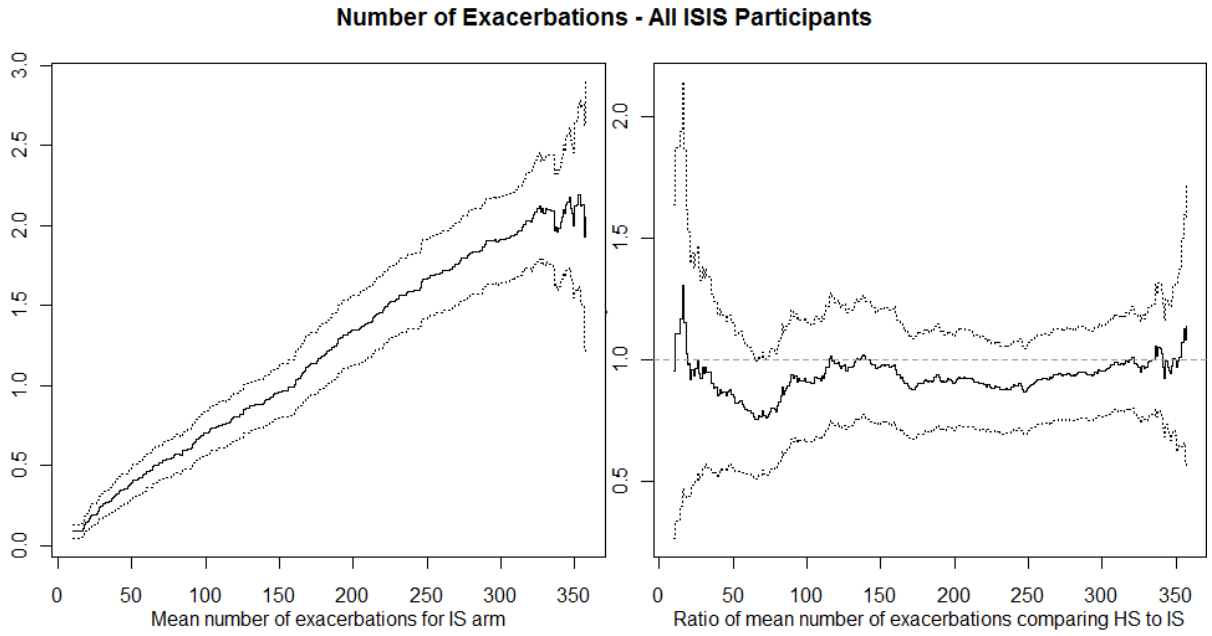

Figure 2: TPR - Total number of exacerbations process

We see that the average number of exacerbations in the IS arm increases in a linear fashion over time since randomization until around the 330 day mark, where participants begin to exit the study (median follow up time was 342 days. Final visits were to occur at week 48 ± 3, or between days 315 and 352 from randomization). This linearity suggests that there were no dramatic changes in the rates of exacerbations over time since randomization in the IS arm. While there appear to be more exacerbations experienced in the HS arm for a brief period at the beginning of the study, followed by a swing in the other direction, the overall behavior of the estimated ratio between arms over time is effectively constant at one. This result is in agreement

with the finding of no treatment effect with respect to exacerbation counts from previous analyses.

The accumulated days in exacerbation process model was fit using the identity link. Coefficient estimates and pointwise 95% confidence intervals for the mean number of accumulated days in exacerbation over time in the IS arm (model intercept) and difference in mean accumulated days in exacerbation over time between HS and IS arms (treatment effect) are presented in Figure 3. Similar to the total number of exacerbations model fit, we see an increasing linear trend in the mean days in exacerbation for the IS arm. This suggests the average individual on IS treatment accrued days in exacerbation in a fairly uniform fashion over time since randomization (again, we see erratic behavior in the estimates once participants begin to exit the study).

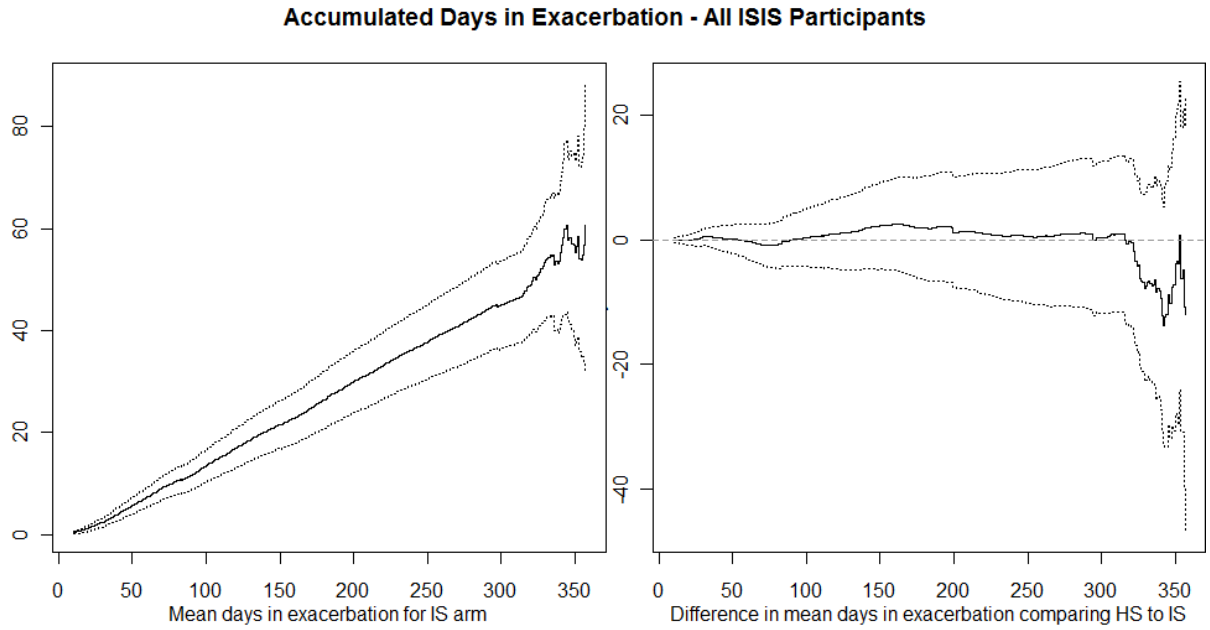

Figure 3: TPR - Accumulated days in exacerbation process

The estimated difference in mean accumulated days in exacerbation between the HS and IS arms is very close to zero for the most of the study duration. Only after day 320 does the point estimate of the difference suggest any benefit for the HS arm, but that is where data begins to become sparse and the pointwise confidence intervals never exclude zero. This result of no treatment effect is consistent with the related finding of Rosenfeld et al. of no difference in antibiotic treatment days between arms (see Section 2.3). [6] The estimated ratio of mean antibiotic treatment days comparing individuals on HS to those on IS was 1.13, 95% C.I. of (0.91, 1.40).

In the context of the entire ISIS trial population we have found similar conclusions to those reported by Rosenfeld et al. in the primary manuscript for the ISIS trial. There is no evidence of an effect of hypertonic saline in reducing occurrences of pulmonary exacerbations based on a number of different methods. We now turn our attention to the iPFT sub-study participants.

5.2. **Infant PFT Sub-study Participants**

With the iPFT sub-study data, we have the ability to include a measure of baseline lung function in our models. One of the lung function measures collected, the forced expiratory volume in one half of a second ($FEV_{0.5}$), is an accepted indicator of overall lung function and health in infants. This in turn is considered an important risk factor for pulmonary exacerbations. Within this subset of participants, we are interested in evaluating both the treatment effect and the effect of baseline $FEV_{0.5}$ on pulmonary exacerbations.

Of the 73 ISIS participants enrolled in the iPFT sub-study, 72 had baseline lung function testing initiated and 61 (85%) of those resulted in acceptable measurements. As per the ISIS trial protocol, acceptability of lung function tests was determined by expert over-reading at the Cystic

Fibrosis Foundation Therapeutics Development Network Infant PFT Resource Center at the University of North Carolina. Acceptability for the purposes of this study was defined in accordance with published guidelines to ensure research quality measurements. While it is concerning that 15% of the iPFT participants did not have acceptable baseline measurements, this data loss was not unexpected given the difficulty of obtaining quality measurements of lung function in infants. For this thesis, we include only the 61 participants with acceptable measurements and ignore the issues with missing data. Ignoring this missingness may lead to bias and increased variability in estimates. [15]

Table 6 contains descriptive statistics for baseline characteristics by arm for the 61 iPFT participants. Baseline characteristics were similar in the two arms.

Table 6: Baseline characteristics by treatment arm for iPFT participants

Characteristic	Hypertonic Saline (n = 29)	Isotonic Saline (n = 32)
Age, mean (SD) (years)	0.7 (0.3)	0.8 (0.3)
Male, No. (%)	13 (45)	18 (56)
Weight, mean (SD), kg	8.3 (1.6)	8.7 (1.5)
Height, mean (SD), cm	68.9 (4.7)	70.3 (5.2)
$FEV_{0.5}$, mean (SD), ml	275.9 (67.6)	282 (66.1)

Table 7 shows the frequency of exacerbation counts by treatment arm for the 61 iPFT participants. Similar to our earlier approach, we will consider collapsed exacerbation strata when modeling an interaction between treatment effect and exacerbation strata, but here we will only use a cutoff of the third exacerbation since data are sparse for counts higher than that.
In addition, modeling effects for treatment and $FEV_{0.5}$, we adjust for sex, age at baseline, and height at baseline because we are interested in the effect of lung function on pulmonary exacerbations among participants of the same gender, age, and height. Despite the strong

correlation between age and height among infants, both are included in these models as age has been observed to be a significant independent predictor of $FEV_{0.5}$ after accounting for height in infants. [16] Mean baseline $FEV_{0.5}$ among the 61 participants was 279 ml with a standard deviation of 66 ml. In the following models, baseline $FEV_{0.5}$ has been divided by the standard deviation (SD) to aid in interpretability, as a one ml difference has little clinical meaning.

Table 7: Frequencies of exacerbations by treatment arm for iPFT participants

Exacerbation Count	0	1	2	3	4	5	6	Total
Isotonic Saline	9	9	4	8	1	1	0	32
Hypertonic Saline	5	10	7	2	1	3	1	29
Total	14	19	11	10	2	4	1	61

5.2.1. **Extensions of the Cox Proportional Hazards Model - iPFT Sub-Study**

The analysis of the time to first pulmonary exacerbation among iPFT participants adjusted for baseline height, age and sex, gives an estimated hazard ratio of 1.23, comparing individuals on HS to those on IS of the same baseline $FEV_{0.5}$ (95% C.I. of (0.67, 2.22)) and an estimated hazard ratio of 1.40 comparing individuals differing in baseline $FEV_{0.5}$ by one SD of the same treatment arm (95% C.I. of (0.84, 2.33)). While this analysis also concludes no treatment effect, the observed hazard ratio is in the opposite direction of that seen in the overall ISIS trial. Kaplan-Meier survival estimates are plotted in Figure 4; however these only estimate the marginal treatment effect and do not take into account $FEV_{0.5}$, height, age, or sex.

The AG, conditional, and marginal models examining an overall treatment effect result in similar conclusions (Table 8), with hazard ratios ranging from 1.05 to 1.10 comparing individuals on HS to those on IS of the same baseline $FEV_{0.5}$, height, age and sex (all confidence intervals contain one).

Again, we see little difference between the marginal and conditional model fits using collapsed exacerbation strata and those that do not, further suggesting that collapsing exacerbation strata may only make sense when estimating treatment effects by exacerbation strata. It is interesting to note that with these models the estimated hazard ratios for the effect of $FEV_{0.5}$ on the risk of pulmonary exacerbation are statistically significant and in the opposite direction of what is expected, with hazard ratios ranging from 1.47 to 2.08 comparing individuals whose baseline

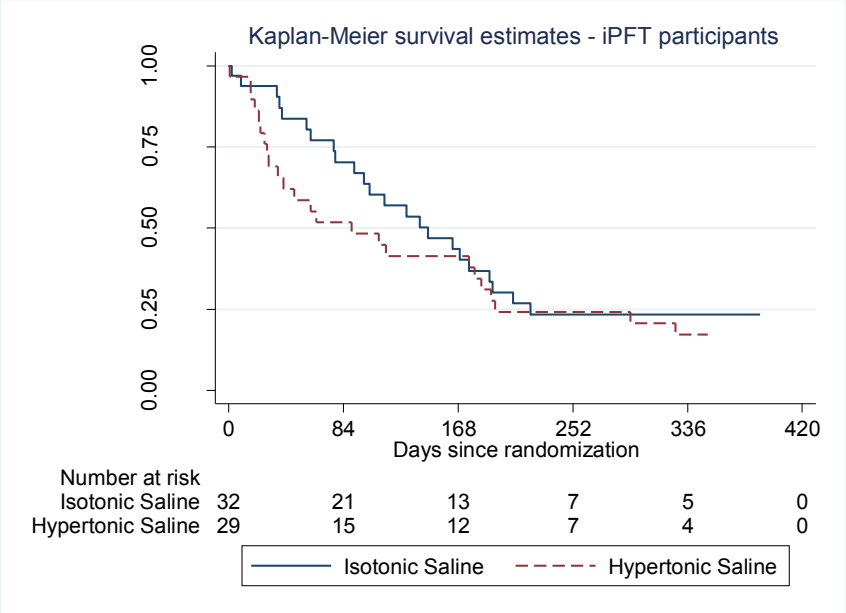

Figure 4: K-M survival estimates by treatment arm for iPFT participants - time to first pulmonary exacerbation

$FEV_{0.5}$ differ by one standard deviation (comparing higher to lower). It is widely thought that a higher $FEV_{0.5}$ for individuals of the same size, age and sex signifies better lung health and should be associated with a lower risk of pulmonary exacerbation, although this expectation is based on studies of older children and adults.

Table 8: Cox model estimates for iPFT participants

Model	Treatment HR (95% C.I.)	$FEV_{0.5}$ (one SD) HR (95% C.I.)
Time to First Exacerbation	1.23 (0.67, 2.22)	1.40 (0.84, 2.33)
Andersen-Gill	1.09 (0.67, 1.78)	1.58 (1.16, 2.14)
Conditional	1.10 (0.73, 1.66)	1.55 (1.16, 2.06)
Conditional – 3 Strata	1.07 (0.71, 1.64)	1.47 (1.12, 1.93)
Marginal	1.06 (0.52, 2.17)	2.08 (1.12, 3.84)
Marginal – 3 Strata	1.05 (0.54, 2.07)	1.97 (1.13, 3.44)

The conditional and marginal models restricted to iPFT participants allowing for a treatment and exacerbation strata interaction effect are summarized in Table 9. These models were also adjusted for baseline height, age, and sex while allowing for a different effect of $FEV_{0.5}$ for each exacerbation strata. Conclusions from the two models are fairly similar with regard to direction and magnitude of treatment effect. The estimated effect of $FEV_{0.5}$ on the risk of exacerbation is also similar between the two models, but the marginal model point estimates are larger in magnitude. The observed direction of effect for $FEV_{0.5}$ is again in the opposite direction of what we might expect, although not statistically significant for the first exacerbation strata in either model.

Table 9: Marginal and conditional models - 3 strata treatment effect for iPFT participants

Model	Treatment HR (95% C.I.)	$FEV_{0.5}$ (one SD) HR (95% C.I.)
Conditional		
First	1.27 (0.70, 2.29)	1.33 (0.89, 1.98)
Second	1.00 (0.50, 2.01)	1.55 (1.09, 2.21)
Third+	0.84 (0.37, 1.90)	1.55 (1.06, 2.27)
Marginal		
First	1.15 (0.62, 2.15)	1.59 (0.91, 2.80)
Second	1.00 (0.44, 2.27)	1.93 (1.06, 3.52)
Third+	0.88 (0.35, 2.24)	2.71 (1.53, 4.80)

5.2.2. **Temporal Process Regression - iPFT Sub-Study**

In this section, we apply TPR to evaluate the effect of treatment and the effect of $FEV_{0.5}$ on pulmonary exacerbations after adjusting for baseline height, age and sex. As in our application of TPR to all ISIS participant data, we will model two processes to encompass the information we have about pulmonary exacerbations: the total number of exacerbations over time process, and the accumulated days in exacerbation over time process.

The total number of exacerbations process model was fit using a log link function. Plots of the exponentiated coefficient estimates and pointwise 95% confidence intervals are presented for the intercept, effect of treatment, and effect of $FEV_{0.5}$ in Figure 5. The model intercept here represents the estimated mean number of exacerbations over time since randomization for males in the IS treatment arm with mean baseline $FEV_{0.5}$ (279 ml), mean baseline age (9.5 months) and mean baseline height (70 cm).

The coefficient plots for sex, baseline height, and age are not presented as they are not the focus of this analysis. The estimated ratios of the mean number of exacerbations over time for age and height are close to one throughout the study for a one month difference in age at baseline or a one centimeter difference in height at baseline, respectively. The estimated ratio over time comparing females to males is between approximately 0.75 and one for most of the study. The corresponding pointwise 95% confidence intervals only exclude a ratio of one for the sex covariate, and that only occurs for brief periods very early and very late in the study (when data are sparse).

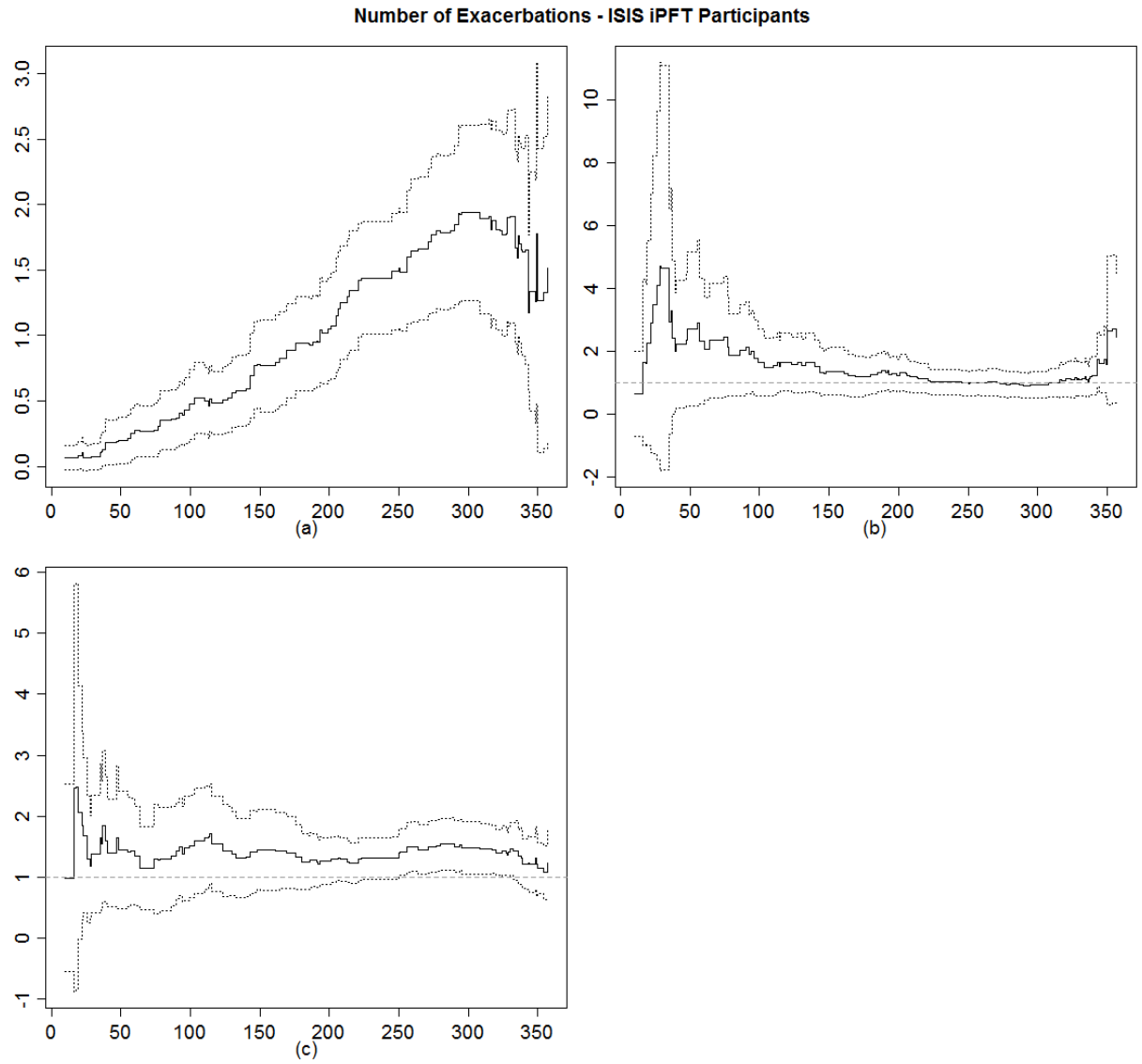

Figure 5: iPFT sub-study TPR - total number of exacerbations process - plots indicate:

(a) The mean number of exacerbations for males in IS arm with mean baseline $FEV_{0.5}$, height, and age
(b) The ratio of the mean number of exacerbations comparing individuals on HS to those on IS of the same sex, baseline $FEV_{0.5}$, height, and age
(c) The ratio of the mean number of exacerbations comparing individuals differing in baseline $FEV_{0.5}$ by one SD of the same treatment arm, sex, baseline height, and age

The estimated ratio over time of mean number of exacerbations for individuals differing in baseline $FEV_{0.5}$ by one SD (of the same baseline height, age and sex) indicates results similar to what was seen in the Cox models. Individuals with higher baseline $FEV_{0.5}$ for their age, height,

sex and assigned to the same treatment were observed to have more exacerbations on average, although for most of the study duration the 95% pointwise confidence intervals did not exclude one.

The accumulated days in exacerbation process model was fit using the identity link function. Plots of the coefficient estimates and pointwise 95% confidence intervals are presented for the intercept, effect of treatment, and effect of $FEV_{0.5}$ in Figure 6. The model intercept here represents the estimated mean number of accumulated days in exacerbation over time since randomization for males in the IS treatment arm with mean baseline $FEV_{0.5,}$ mean baseline age and mean baseline height.

The coefficient plots for sex, baseline height, and age are not presented. The estimated difference in mean days in exacerbation over time for a one month difference in age (comparing older to younger) increases linearly from zero days to 5 additional days at 320 days from randomization where participants begin to leave the study. The estimated difference in mean days in exacerbation over time for a one centimeter difference in height at baseline (comparing taller to shorter) steadily decreases from zero to -4 days at 320 days. Finally, the estimated difference in mean days in exacerbation for females compared to males is approximately zero throughout. The corresponding pointwise 95% confidence intervals do not exclude zero except those for the height covariate during a 5-10 day period near the end of the study.

Much like the total number of exacerbations process fit for the iPFT sub-study, the expected number of accumulated days in exacerbation increases linearly over time for these participants.

This suggests that, on average, they not only experience exacerbations across the study duration at a fairly constant rate, but the days spent exacerbated accrue uniformly.

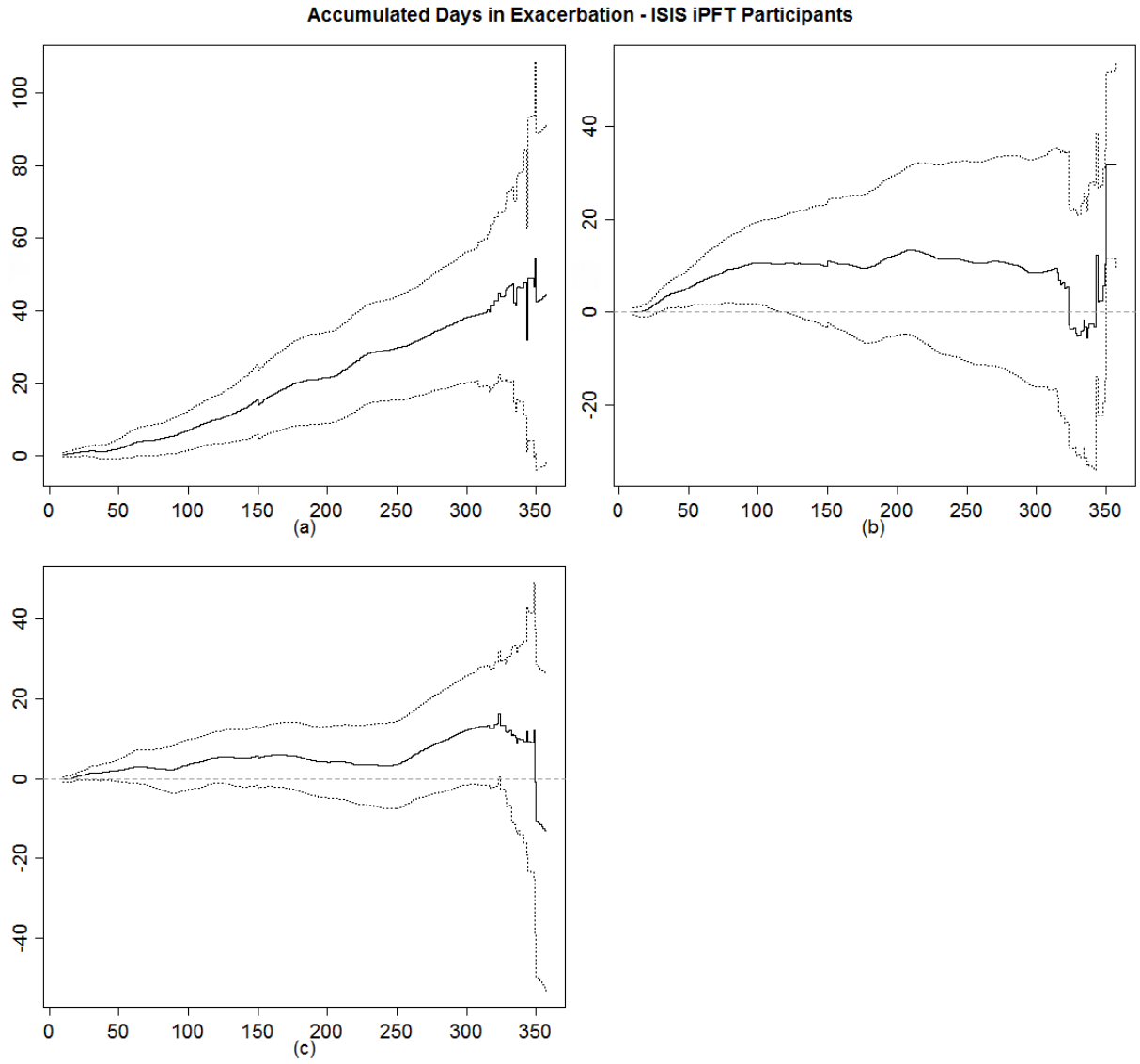

Figure 6: iPFT sub-study TPR - accumulated days in exacerbation process - plots indicate:

(a) The mean days in exacerbation for males in IS arm with mean baseline $FEV_{0.5}$, height, and age
(b) The difference in mean days in exacerbation comparing individuals on HS to those on IS of the same sex, baseline $FEV_{0.5}$, height, and age
(c) The difference in mean days in exacerbation comparing individuals differing in baseline $FEV_{0.5}$ by one SD of the same treatment arm, sex, baseline height, and age

The estimated difference in mean accumulated days in exacerbation between the HS and IS arms (for participants of the same baseline $FEV_{0.5}$, height, age and sex) increases for the first 100 days of the study before remaining at approximately ten additional days in exacerbation for HS individuals for the rest of the duration. This pattern is congruent with the observation of more exacerbations in the HS arm iPFT sub-study participants early in the study if exacerbations are of approximately the same duration (i.e. those ten additional days in exacerbation on average were accumulated early and persisted as the only real difference between treatment arms over the rest of the study duration). Early pointwise 95% confidence intervals exclude a mean difference of zero, but over time that difference dissolves. Participants with higher baseline $FEV_{0.5}$ for their age, height, sex and assigned to the same treatment spent more time in exacerbations on average although for most of the study duration the 95% pointwise confidence intervals did not exclude zero.

6. **Data Construction and Simulation**

We have now seen the study conclusion of no hypertonic saline treatment effect from the Rosenfeld et al. analyses confirmed using extensions of the Cox proportional hazards model as well as with temporal process regression, not only in the overall ISIS study population, but also in the iPFT sub-study. While TPR did not provide different conclusions from the other methods, it did serve to give us a more complete picture of pulmonary exacerbation behavior in the ISIS clinical trial. As such, TPR appears to be a useful tool for examining data from a recurrent episode setting involving events of the same type with durations that may last for some time.

In the following sections our goal is to investigate whether it is possible to generate recurrent episode data such that Poisson regression and the simple Cox model (i.e. one considering first events only) cannot detect a difference, while the extensions of the Cox model and TPR can. We begin by presenting the results of applying these methods to one such generated data set as a proof-of-concept example. We then discuss the steps used to construct this dataset. Finally, we conclude by reproducing some recurrent event simulations to demonstrate a future direction for this investigative work.

6.1. **Constructed Dataset Results**

In this section we use a constructed dataset to show that extensions of the Cox proportional hazards model and temporal process regression may detect evidence of a treatment effect that goes undetected by Poisson regression and simple Cox model analyses. To be similar to the ISIS trial, data were simulated for 150 subjects per treatment arm with exactly 48 weeks of follow up time. These data were generated to have similar times to first events, rates of all events and event durations between the two treatment arms as was seen in ISIS. Additionally, the accrual of events after the first was constructed to start slowly in the treatment arm and increase over time, while in the control arm it was constructed to start fast and decrease over time. See Section 6.2 for details on this construction. This discrepancy between arms in timing of events is expected to be detected by extensions of the Cox model and TPR.

Our simulation resulted in a dataset with event rates of 2.3 per person-year in the control arm and 2.4 per person-year in the treatment arm. Table 10 summarizes the frequencies of event counts by treatment arm. The unadjusted Poisson regression model used by Rosenfeld et al. estimated a

rate ratio comparing treatment to control of 1.04 with a 95% C.I of (0.90, 1.22). A Cox proportional hazards model of the time to first event produced an estimated hazard ratio comparing treatment to control of 0.96 with a 95% C.I. of (0.75, 1.22). The corresponding Kaplan-Meier survival curves are plotted in Figure 7. As intended, these analyses suggest no difference between the two simulated treatment arms.

Table 10: Frequencies of events by treatment arm for constructed data

Event Count	0	1	2	3	4	5	6	7	Total
Control	17	34	42	35	17	4	1	0	150
Treatment	20	43	24	32	15	13	2	1	150
Total	37	77	66	67	32	17	3	1	300

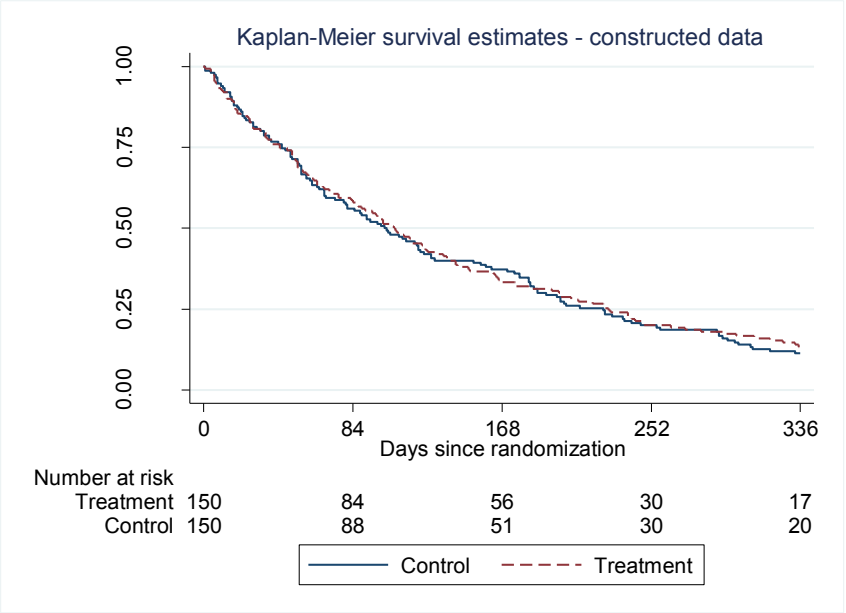

Figure 7: K-M survival estimates by treatment arm for constructed data - time to first event

Estimated hazard ratios from the Cox proportional hazards model of time to first event as well as the extensions of the Cox proportional hazards model estimating an overall (or common) treatment effect are presented in Table 11. Conditional and marginal models are applied using the exacerbation information "as is," as well as using cutoffs of the third and fifth event as

described in Section 5.1. As was seen with the ISIS data, the choice of number of event strata (referred to as exacerbation strata previously) used does not appear to make much difference when estimating a common treatment effect. While these models consider all events beyond the first, they do not suggest any differences between the treatment arms. This is not unexpected, as the common treatment effect estimates aggregate information across event strata and we know that on average both the rate of events and the timing of first events is alike between arms.

Table 11: Cox model treatment effect estimates - constructed data

Model	Treatment Effect	p-value
Time to First Event	0.96 (0.75, 1.22)	0.736
Andersen-Gill	1.05 (0.88, 1.24)	0.613
Conditional	1.10 (0.93, 1.31)	0.270
Conditional – 3 Strata	1.12 (0.94, 1.33)	0.200
Conditional – 5 Strata	1.11 (0.93, 1.32)	0.252
Marginal	0.90 (0.70, 1.16)	0.426
Marginal – 3 Strata	0.92 (0.72, 1.18)	0.507
Marginal – 5 Strata	0.90 (0.70, 1.16)	0.426

Similar to our analyses in Section 5.1.1, we consider treatment and event strata interactions for both the marginal and conditional models using three and five event strata. The results from the three strata models are presented in Table 12. Here we see the first evidence that the differences in the timing of event accrual are being identified. In both of these models the estimated hazard ratios for second events suggest a statistically significant beneficial treatment effect. Also as expected, the direction of the estimated hazard ratios for events beyond the second is in favor of the control arm, but only the conditional model indicates this as statistically significant.

These results are consistent with behavior of the marginal model noted by Kelly and Lim in a 2000 paper; they observed that marginal models exhibit “carry over effects” from preceeding

strata. [17] In other words, since there is no treatment effect for the first event, the marginal model carry over effect results in the hazard ratio estimate for the second event being drawn closer to one. Similarly, any evidence that events beyond the second are occurring more frequently in the treatment arm is attenuated by the carry over effect from the estimate directions for the first and second events. We also see potential evidence of this carry over effect in the direction of the point estimates for the common treatment effect marginal models.

Table 12: Marginal and conditional models - 3 strata treatment effect - constructed data

Marginal	Treatment HR (95% C.I)	p-value	Conditional	Treatment HR (95% C.I)	p-value
First	0.96 (0.75, 1.22)	0.736	**First**	0.96 (0.75, 1.22)	0.736
Second	0.65 (0.49, 0.87)	0.004	**Second**	0.51 (0.38, 0.68)	<0.0005
Third+	1.23 (0.86, 1.76)	0.255	**Third+**	3.22 (2.37, 4.37)	<0.0005

The marginal and conditional models with five treatment and event strata interactions are summarized in Table 13. As expected, the estimates for the first two events are identical to corresponding estimates from the three strata models. However, starting with the third event we see a further departure between the two model's estimates. The conditional model hazard ratios suggest the treatment group has a higher risk of third and fourth events as well as any events beyond the forth compared to control, conditional on having had the preceding events. Additionally, all of these higher strata estimates are statistically significantly different from the null hazard ratio of one. In contrast, the marginal model suggests no difference between arms in the risk of a third event, and while the direction of the estimated effects for fourth events and any events beyond that suggest that treatment is not beneficial, the confidence intervals are wide and do not exclude one.

A Correction for multiple comparisons should be made here as well as for the models in Table 12, but the choice of method to account for this is not obvious. P-values are provided to allow the reader to more easily adjust for multiple comparisons as they see appropriate. With a Bonferroni correction applied to the models with three strata (dividing the 0.05 significance level by 3) and to the models with five strata (dividing the 0.05 significance level by 5), the conclusions with regard to statistical significance of treatment effects are unchanged.

Table 13: Marginal and conditional models - 5 strata treatment effect - constructed data

Marginal	Treatment HR (95% C.I)	p-value	Conditional	Treatment HR (95% C.I)	p-value
First	0.96 (0.75, 1.22)	0.736	**First**	0.96 (0.75, 1.22)	0.736
Second	0.65 (0.49, 0.87)	0.004	**Second**	0.51 (0.38, 0.68)	<0.0005
Third	0.98 (0.68, 1.41)	0.907	**Third**	2.38 (1.56, 3.62)	<0.0005
Fourth	1.42 (0.81, 2.50)	0.218	**Fourth**	3.96 (2.22, 7.08)	<0.0005
Fifth+	5.55 (0.53, 58.46)	0.153	**Fifth+**	8.76 (2.73, 28.00)	<0.0005

The time-varying coefficient plots from fitting the total number of events process and the accumulated days in episodes process TPR models are presented in Figure 8 and Figure 9, respectively. The total number of events process model was fit using a log link function. The corresponding plots display exponentiated coefficient estimates and pointwise 95% confidence intervals indicating the mean number of events over time in the control arm (left) and the ratio of the mean number of events over time comparing treatment and control (right).

The estimated mean number of events over time in the control arm is linear over the first 200-250 days with perhaps a slight decrease in slope over the last 100 days. This is somewhat unexpected, given the timing of event accrual we constructed for the control arm (starting fast and decreasing over time) we would expect there to be a period early on where the mean number of events in the control arm grows quickly, especially in contrast with the later periods where

control arm events accumulate more slowly. If that expectation is correct, the trajectory followed by the coefficient estimate for the mean number of events should be increasing with a decreasing slope over time.

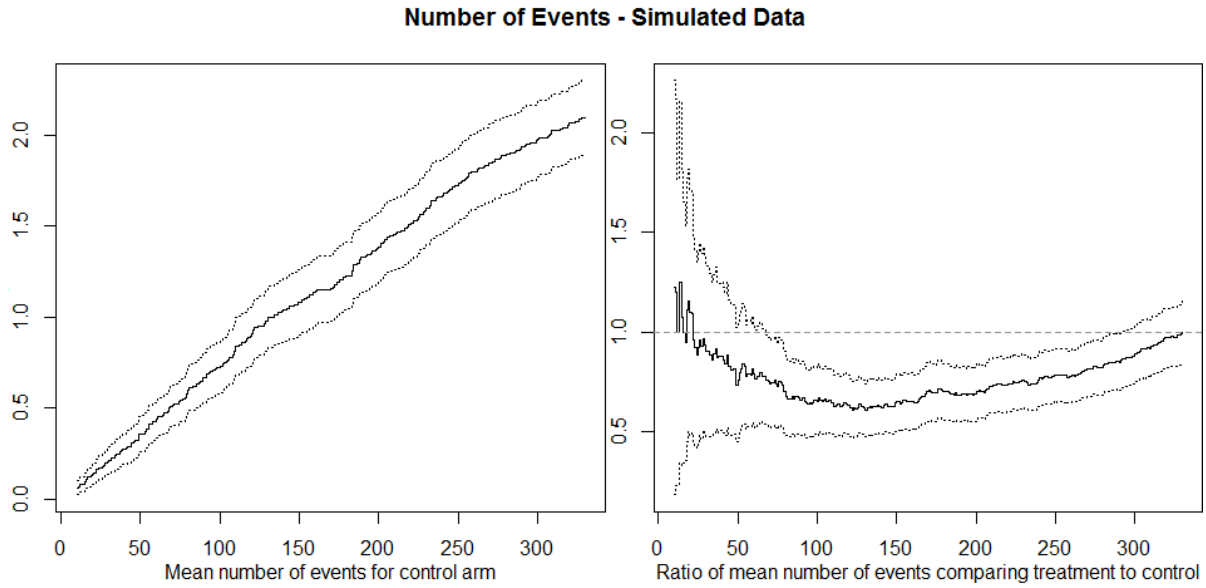

Figure 8: TPR - total number of events process - constructed data

The estimated ratio of mean events between the two arms suggests differences in the timing of events between the arms, with early estimates indicating no difference, followed by a period favoring the treatment arm until roughly 150 days when the ratio starts to increase towards one. In other words, we have constructed a dataset that exhibits no difference in ratio of mean number of events between treatment groups both early in the study and at the conclusion, as well as no difference in the risk of first event, but during the majority of the trial a beneficial treatment effect is suggested with pointwise 95% confidence intervals that exclude a ratio of one.

The accumulated days in episodes process model was fit using the identity link function. The corresponding plots display the coefficient estimates and pointwise 95% confidence intervals

indicating the mean number of accumulated days in episodes over time in the control arm (left) and difference in mean accumulated days in episodes over time between treatment and control arms (right).

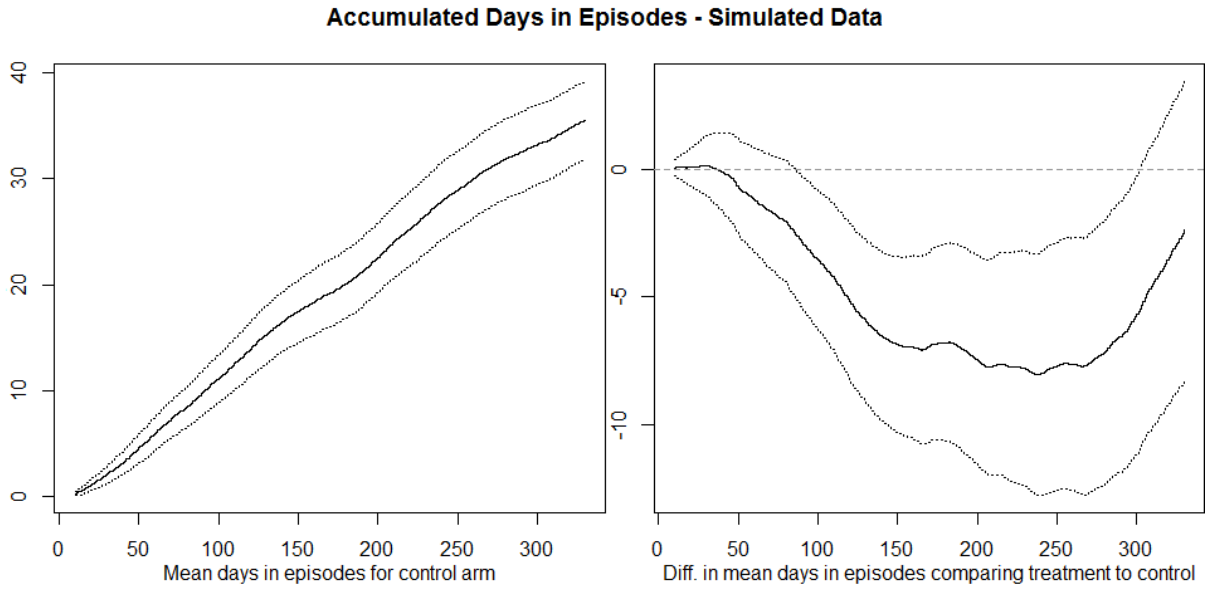

Figure 9: TPR - accumulated days in episodes process - constructed data

Given that the event durations were simulated under the same distribution for both arms, it is not surprising that the behavior of these plots follows that of the mean number of events closely. As with the mean number of events, the time-varying coefficient estimates for the difference in days in episodes suggest no difference at the beginning and end of the study with a large period in the middle of significantly fewer days in episodes for the treatment arm.

6.2. **Construction of Datasets from Non-homogeneous Poisson Processes**

Within this section we detail the framework used for generating datasets like the one just presented. As previously mentioned, we want to construct data such that there are about 150 subjects per arm with approximately 48 weeks of follow up. Additionally, both ISIS treatment

arms had nearly identical times to first events, an average rate of 2.3 events per person-year (or 2.1 per 48 person-weeks), and an average event duration of 15 days, so we seek to mirror those characteristics.

Our goal is to produce data such that the conclusions of the extensions of the Cox model and TPR are not consistent with null results from the analyses used by Rosenfeld et al. that look at the average rate of events (Poisson regression) or the timing of first events (simple Cox model). Since the rate of events and the timing of first events must be similar between arms, the only component of recurrent event data left to manipulate is the timing of events after the first. As such, creating differences in the rates at which those events are accrued across the study duration will be the focus of our data construction.

The results of analyzing the ISIS data suggest events in the trial were accumulated at a fairly constant rate (Figure 2 from the TPR analysis illustrates this). We need to introduce differences between arms in the accrual of events after the first, but simply fixing one arm at a lower event rate and one at a higher rate will not satisfy our requirement of equivalent rates between arms. Therefore, we need a way to induce non-constant rates of event accrual (after the first event) over the study duration.

One way to induce such patterns is to use time-dependent event-rate functions for events after the first, for example, the treatment arm event-rate function could start out low and increase over time and the control arm event-rate function could start out high and decrease over time. If two such functions can be specified so that the overall rate of events is similar over the study

duration, this construction would result in data that satisfies our two requirements but has varying rates of event accrual throughout the study. In other words, we would expect the risk of a first event to be identical across treatment arms, but then the risk of subsequent events would be higher in the control arm for some length of time, followed by an interval of similar risk in the two arms, and finally a period where the risk is higher in the treatment arm.

An approach to generating data of this type is to use a non-homogeneous Poisson process with time varying rate (or intensity) functions. The R package '`NHPoisson`' provides some functionality for doing just that, it allows the specification of time-dependent intensity functions over a given time period. [18,14] This is what we desire, except that we also require the first events to be generated according to a constant intensity function. To accomplish this we need the event generating mechanism to switch between a constant intensity for first events, and time-dependent (non-constant) intensity for all subsequent events. Unfortunately, the package does not allow specification of intensity functions in this "event-dependent" manner.

To get around this limitation we will use the '`NHPoisson`' package to simulate two datasets of 300 individuals (150 in each arm) over a 336 day duration (48 weeks). One dataset will use a homogeneous Poisson process with a constant intensity of 2.3 events per person-year for both treatment arms. The second dataset will use a non-homogeneous Poisson processes with two different time-dependent intensity functions (one for each arm). These intensity functions are rescaled versions of the functions $f(t) = t^2$ and $g(t) = (t - 336)^2$ for the treatment and control arm, respectively. Once these two datasets are generated they will be combined by taking the times to first events from the homogenous/constant intensity dataset and appending the times

to events from the non-homogeneous dataset to create the times to all subsequent events. This combined dataset will be administratively censored at 336 days. For example, individuals with no first event from the constant intensity dataset will have no events through the entire 336 day duration in the combined dataset.

If we could use one Poisson process with an entirely specified intensity function, it would be straight forward to compute the correct rescaling factors for the intensity functions $f(t)$ and $g(t)$ as the expected number of events from a non-homogenous Poisson process is just the integral of the intensity function over the entire time interval. The need to combine the two datasets as described above makes the analogous computation extremely challenging, if not impossible. Instead, the rescaling factors for the two time-dependent intensity functions were determined using an iterative process of simulating 100 datasets of 300 subjects to estimate the true rate of events in the two arms, then updating the rescaling factors to approach the target rate of 2.3 events per person-year.

Finally, event durations are simulated separately and overlaid on top of the time to event data generated for each subject. If an event's duration causes an overlap with a later event for a subject, the two events are combined and considered one event with a duration that covers the time from the start of the first event until the end of the later event (i.e. the durations were not simply added together). This process of combining overlapping events is consistent with what was done for the ISIS trial when a subject had two (or more) overlapping courses of antibiotic treatment for sets of symptoms that were described as not related on study forms. The event durations were all considered independent and were simulated using a Weibull distribution with

parameters $\alpha = 2$ and $\beta = 17$ where α and β are the shape and scale parameters, respectively. This distribution was chosen as it mimics the empirical exacerbation duration distribution of the ISIS trial fairly well, with the exception of some of the particularly lengthy exacerbations.

Unfortunately, since these data are not generated from some known truth under the Poisson regression model, extensions of the Cox proportional hazards model, or temporal process regression model, true simulation study type results are not possible. Additionally, repeated application of the TPR methods does not yield easily summarized results across a number of datasets. In Section 6.1 we saw the results of applying our analyses to one dataset constructed using this framework. See Appendix D for code corresponding to construction of this data as well as application of the analyses.

6.3. **Reproduction of Kelly and Lim Recurrent Event Simulations**

As a conclusion to this investigation using constructed datasets, we reproduce a portion of the recurrent event simulation studies carried out by Kelly and Lim in 2000. [17] These simulation studies demonstrate a desired future direction for the work presented thus far in Section 6. Kelly and Lim simulated datasets under the framework of recurrent events without duration, with a maximum of four events per participant. They considered two scenarios: one with a constant treatment effect across all four events, and one where treatment is only effective for the first event.

Event times t_{ik} were generated using the mechanism $t_{ik} = U_{ik} * e^{(\beta_0 + \beta_k x_i)}$ where U_{ik} is an exponential random variable with mean 1, $\beta_0 = -3$ in all scenarios, $\beta_k = -1$ for $k = 1, 2, 3, 4$ under

scenario (i) and $\beta_1 = -1$, $\beta_2 = \beta_3 = \beta_4 = 0$ under scenario (ii), and x_i is a binary treatment indicator. Each simulated dataset consisted of 500 subjects with 250 receiving treatment. Administrative censoring was imposed after 120 days, that is, a subject's events were censored if the additive total time since the start of the study exceeded 120 days. Kelly and Lim fit the AG and conditional extension of the Cox proportional hazards model to the data from 100 repeated simulations under both scenarios. Their application of the conditional model was first with a common treatment effect estimate and then with a treatment event interaction, similar to our work with the ISIS data.

The results from replicating this simulation study are presented in Table 14 and Table 15. Our findings are similar to those of Kelly and Lim. As expected, the mean estimates of the common treatment effect, β, under scenario (i) using the AG and conditional models are close to -1 (-1.003 and -0.996) when the truth is $\beta = -1$. Also as expected, the mean estimates of the event specific treatment effect βs under scenario (ii) using the conditional model are close to the truth (-1.010, -0.101, 0.009 and -0.013) when the true βs are -1, 0, 0 and 0, for the first, second, third and fourth event, respectively.

Table 14: Recurrent event simulation results - Kelly & Lim scenario (i)

	$\boldsymbol{\beta}$	Mean($\widehat{\boldsymbol{\beta}}$)	SD($\widehat{\boldsymbol{\beta}}$)	Bias	Mean estimated NSE($\widehat{\boldsymbol{\beta}}$)*	Mean estimated RSE($\widehat{\boldsymbol{\beta}}$)*	Coverage naïve 95% C.I.	Coverage robust 95% C.I.
AG	-1.0	-1.003	0.055	-0.003	0.056	0.056	0.96	0.96
Conditional	-1.0	-0.996	0.060	0.004	0.061	0.061	0.95	0.95
Conditional								
1st event	-1.0	-1.002	0.091	-0.002	0.099	0.100	0.97	0.96
2nd event	-1.0	-1.003	0.125	-0.003	0.112	0.112	0.93	0.94
3rd event	-1.0	-0.984	0.130	0.016	0.133	0.132	0.94	0.94
4th event	-1.0	-0.990	0.153	0.010	0.172	0.171	0.96	0.96

* NSE and RSE are the naive and robust standard errors, respectively

Table 15: Recurrent event simulation results - Kelly & Lim scenario (ii)

	β	Mean($\hat{\beta}$)	SD($\hat{\beta}$)	Bias	Mean estimated NSE($\hat{\beta}$)*	Mean estimated RSE($\hat{\beta}$)*	Coverage naïve 95% C.I.	Coverage robust 95% C.I.
AG	-0.25	-0.427	0.056	-0.177	0.049	0.058	0.05	0.13
Conditional	-0.25	-0.270	0.052	-0.020	0.052	0.052	0.93	0.93
Conditional								
1st event	-1.00	-1.010	0.100	-0.010	0.100	0.099	0.94	0.93
2nd event	0.00	-0.010	0.095	-0.010	0.099	0.098	0.94	0.94
3rd event	0.00	0.009	0.104	0.009	0.101	0.101	0.94	0.94
4th event	0.00	-0.013	0.099	-0.013	0.108	0.107	0.96	0.97

* NSE and RSE are the naive and robust standard errors, respectively

The large bias and poor coverage exhibited by the AG model under scenario (ii) was also seen by Kelly and Lim but they did not provide an explanation. However, the common effect conditional model applied to the same data does not exhibit these problems, so the explanation is likely in the differences between these models. Unlike the conditional model, the AG model does not estimate a common treatment effect by pooling stratified estimates. In fact, it does not differentiate between event strata at all, which leads to the differences from the first event treatment effect being overemphasized for the common treatment effect estimate. The code used in these simulations is available in Appendix E.

The end goal here would be to implement standard simulation studies with datasets similar to our constructed dataset from Section 6.1. Unfortunately it does not appear that the framework established by Kelly and Lim is easily adapted to satisfy some of the requirements of those datasets. For example, while ensuring similarity between times to first events in the treatment arms is possible, attaining similar event rates over the study duration while introducing different treatment effects across events is a challenge. Additionally, we need to be able to simulate event

durations and would like to not explicitly restrict the total number of events per subject (as a reminder, some ISIS participants had as many as 9 pulmonary exacerbations).

7. Future Directions and Conclusions

Our simulation and data construction work leaves a few intriguing questions unanswered, such as: Can recurrent event or episode data be generated in other ways such that Poisson regression and the simple Cox model cannot find statistically significant differences in treatments, but extensions of the Cox model or TPR can? What are the statistical ramifications of collapsing event data into lower strata as was done here (as suggested by Therneau and Grambsch)? It would also be of interest to explore scenarios where event durations vary with time, or where there is a treatment effect that only impacts episode durations.

Through our exploration of the iPFT sub-study we have uncovered some evidence suggesting that higher baseline lung function is associated with higher risk and/or rates of pulmonary exacerbations in infants with cystic fibrosis. This evidence warrants further exploration and thought, as it goes against the literature on the relationship between lung function and pulmonary exacerbations among older children and adults.

The application of both the extensions of the Cox proportional hazards model and the novel TPR methodology to data from the ISIS clinical trial did not provide any conclusions inconsistent with those from the original analyses of Rosenfeld et al. It is evident through our simulation work that some of these additional analyses (namely those with a treatment and event strata interaction or TPR) may reveal behavior undetected by simpler models. We have also produced

additional evidence against the use of marginal models in situations with recurrent events or episodes of the same type.

It is our conclusion that in a recurrent episode setting, use of TPR or extensions of the Cox proportional hazards model should be considered over the more traditional Poisson regression and simple Cox model when there is reason to believe that the risk of events varies across event counts. If episode durations are a primary interest, TPR can be an effective tool in assessing the average time spent in episodes over time.

8. **References**

1. Fuchs, H. J. et al., Effect of Aerosolized Recombinant Human DNase on Exacerbations of Respiratory Symptoms and on Pulmonary Function in Patients with Cystic Fibrosis. New England Journal of Medicine 331 (10), 637-42 (1994).
2. Yan, J. & Fine, J. P., Analysis of Episodic Data with Applications to Recurrent Pulmonary Exacerbations in Cystic Fibrosis Patients. Journal of the American Statistical Association 103 (482), 498-510 (2008).
3. Ramsey, B. W. et al., Intermittent Administration of Inhaled Tobramycin in Patients with Cystic Fibrosis. The New England Journal of Medicine 340 (1), 23-30 (1999).
4. Saiman, L. et al., Effect of Azithromycin on Pulmonary Function in Patients with Cystic Fibrosis Uninfected with Pseudomonas aeruginosa. Journal of the American Medical Association 303 (17), 1707-1715 (2010).
5. Elkins, M. R. et al., A Controlled Trial of Long-Term Inhaled Hypertonic Saline in Patients with Cystic Fibrosis. The New England Journal of Medicine 354 (3), 230-240 (2006).
6. Rosenfeld, M. et al., Inhaled Hypertonic Saline in Infants and Children Younger Than 6 Years With Cystic Fibrosis. Journal of the American Medical Association 307 (21), 2269-2277 (2012).
7. Hosmer, D. W., Lemeshow, S. & May, S., Applied Survival Analysis (John Wiley & Sons, Inc., Hoboken, New Jersey, 2008).
8. Therneau, T. M. & Grambsch, P. M., Modeling Survival Data: Extending the Cox Model (Springer-Verlag, New York, 2000).
9. Therneau, T. M. & Hamilton, S. A., rhDNase as an Example of Recurrent Event Analysis. Statistics in Medicine 16, 2029-2047 (1997).
10. Andersen, P. K. & Gill, R. D., Cox's Regression Model for Counting Processes: a Large Sample Study. The Annals of Statistics 10 (4), 1100-1120 (1982).
11. Wei, L. J., Lin, D. Y. & Weissfeld, L., Regression Analysis of Multivariate Incomplete Failure Time Data by Modeling Marginal Distributions. Journal of the American Statistical Association 84 (408), 1065-1073 (1989).
12. Prentice, R. L., Williams, B. J. & Peterson, A. V., On the Regression Analysis of Multivariate Failure Time Data. Biometrika 68 (2), 373-379 (1981).
13. Fine, J. P. & Kosorok, M. R., Temporal Process Regression. Biometrika 91 (3), 683-703 (2004).
14. R Development Core Team, R: A language and environment for statistical computing. (R Foundation for Statistical Computing, Vienna, Austria, 2011).
15. Little, R. J. A. & Rubin, D. B., Statistical Analysis with Missing Data (Wiley, Hoboken, New Jersey, 2002).
16. Jones, M. et al., Forced Expiratory Flows and Volumes in Infants. American Journal of Respiratory and Critical Care Medicine 161, 353-359 (2000).
17. Kelly, P. J. & Lim, L. L.-Y., Survival Analysis for Recurrent Event Data: An Application to Childhood Infectious Diseases. Statistics in Medicine 19, 13-33 (2000).
18. Cebrian, A. C., NHPoisson: Modeling and validation of non homogeneous Poisson processes, R package version 2.0., http://CRAN.R-project.org/package=NHPoisson (2013).

9. **Appendix A - Pulmonary Exacerbation (PEX) Definition:**

A protocol defined pulmonary exacerbation for the ISIS trial occurred when:

"A participant was being treated with oral, inhaled, or intravenous antibiotics AND fulfillment of one or more of the criteria listed below, within the period 3 days prior to antibiotic start date through antibiotic stop date:

Criteria:

(1) Oxygen saturation <90% on room air *or* ≥ 5% decline from previous baseline
(2) New lobar infiltrate(s) or atelectasis on chest radiograph
(3) Hemoptysis (more than streaks on more than one occasion in past week)
(4) Increased work of breathing or respiratory rate
(5) Increased cough
(6) Worked harder than usual to breathe during physical activity
(7) Increased chest congestion or change in sputum
(8) New or increased adventitial sounds on lung exam
(9) Weight loss ≥5% of body weight or decrease across 1 major percentile in weight percentile for age in past 6 months"

Participants were considered to be in a pulmonary exacerbation episode while on any exacerbation defining course of antibiotics (see above) or while on a course of antibiotics that was either denoted to be a continuation of an exacerbation defining antibiotic course (this was an option on ISIS study forms) or denoted to have preceded an exacerbation defining antibiotic course (i.e. earlier treatment for symptoms that later satisfied the exacerbation definition). In other words, participants were considered to have started an exacerbation episode before they satisfied the criteria above if they were later prescribed antibiotics for symptoms that satisfied the above definition and were also considered a continuation of the symptoms for which the first antibiotic course was prescribed. Participants were also not considered to be out of an exacerbation episode until after they were no longer taking prescribed antibiotics for the exacerbation defining symptoms (even if those symptoms no longer satisfied the definition). Finally, if a participant was on two or more exacerbation defining courses of antibiotics that

overlapped at all in duration, the participant was considered to have had one exacerbation that lasted from the earliest antibiotic course start date until the latest antibiotic course stop date.

Notes on the PEX definition:

The PEX definition used for the ISIS trial allows for an exacerbation to start on the same day a previous exacerbation ended. The existing methods (AG, conditional, marginal) ignore such exacerbations as the methods do not allow events to occur for individuals without any time at risk. To force these methods to count these events (of which there are 12) one half day of time has been added between the exacerbations. Similarly the ISIS PEX definition allows exacerbations to begin on the day of enrollment and the existing methods view this as an event without any time at risk. One half day of time has also been added in these cases (of which there are 3) so that they are included in the analyses. All of these added half days are also included for the TPR analyses presented here though they are not required by the model.

10. **Appendix B - rhDNase Analyses Verification:**

10.1. **Therneau and Grambsch:**

Verification of Therneau and Grambsch's analysis of the rhDNase dataset presented in their book: Modeling Survival Data: Extending the Cox Model was done using the rhDNase dataset provided on Terry M. Therneau's website on their rhDNase for Cystic Fibrosis example (http://www.mayo.edu/research/documents/dnasehtml/DOC-10026754) and Stata/IC Version 11.2.

Verification of results presented in their book was almost exact. Different results of some of these same analyses were also presented in Therneau and Hamilton's 1997 paper: rhDNase as an Example of Recurrent Event Analysis, this discrepancy is addressed on Therneau's website for the rhDNase trial:

"To our embarrassment, we cannot exactly reproduce the numbers in the paper. There are multiple ways to define an infection, the number of endpoints and exact timing of them changed at times during the analysis, and we didn't save copies of the relevant data. (For instance, does an infection start with oral antibiotic or is IV antibiotic required?) None of the substantive conclusions is changed; this data set gives the results in the book."

Additionally, the rhDNase dataset includes a small number of records from individuals who were recorded to have exacerbations that started before randomization. Two of these cases persist for the duration of the study and so these individuals are excluded from analysis entirely. The remaining four cases of pre-randomization exacerbations end during the study and in Therneau and Grambsch's analysis these individuals are included following the conclusion of their pre-randomization exacerbation (which is not counted as an event for purposes of analysis). In Therneau's words: "A few subjects were infected at the time of enrollment, [participant] 951317

for instance has a first infection interval of [day] -21 to 7. We do not count this first infection as an 'event', and the subject first enters the risk set at day 7."

10.2. **Yan and Fine (TPR):**

Verification of Yan and Fine's analysis of the rhDNase dataset using their temporal process regression (TPR) methodology was done using R 2.13.0. Both example code and dataset were obtained from the documentation for the '`tpr`' package for R. A number of modifications were made to the provided example code:

1. The FEV variable provided matches the variable in Therneau and Grambsch's dataset once the specified changes are made – that is once it is "centered by its mean and then divided by ten," but intercept coefficient estimates from TPR (corresponding to individuals with mean FEV) models only match those presented when FEV rescaled but not centered.

2. Treatment of pre-randomization exacerbations: verification results most closely matched those presented when records associated with pre-randomization exacerbation were simply deleted from the included dataset. This action leads to some odd implications for the six individuals with pre-randomization exacerbations:

    - Two of these individuals had pre-randomization exacerbations persist for the duration of the study, so these individuals are treated appropriately here (i.e. excluded from analysis).

    - Two other individuals had only a pre-randomization exacerbation that ends during the study, these individuals are excluded entirely from the analysis with this approach, which seems inappropriate (as noted above, Therneau and Grambsch included their observation time after the pre-randomization exacerbation ended).

    - The final two individuals had a pre-randomization exacerbation as well as at least one exacerbation while on study. These individuals are almost treated appropriately here; the only issue is that they are considered at risk for an exacerbation starting at randomization instead of, for example, after their pre-randomization exacerbation ended.

3. Finally, the rhDNase trial exacerbation definition requires individuals have a period of 6 days exacerbation-free after an exacerbation ends before they are at risk for another, separate, exacerbation. Therneau and Grambsch note this is their analyses and make corrections for

this in example SAS code on provided on Therneau's website but the example code for the TPR method does not appear to account for this time and it is not already accounted for in the provided dataset.

Despite these modifications and agreement in the coefficient plots in Figure 4, the integral test statistics presented in the paper for the rhDNase data (Table 2) were not able to be verified (before or after the above modifications were made). Dr. Yan was contacted via email and then phone in April of 2013 and was also unsure of the reason for this discrepancy.

10.3. **Therneau and Grambsch Stata Verification Code:**

```
** Stata code to verify the results presented on pages 214-215 of Therneau &
** Grambsch (Modeling Survival Data: Extending the Cox Model, 2000)

** Read in the CSV dataset available on Therneau's rhDNase trial website:
** http://mayoresearch.mayo.edu/mayo/research/biostat/upload/therneau_upload
** /dnase.html
** This dataset has a number of records for each subject equal to the total
** number of exacerbations they experienced during the trial, and one record
** for each subject without an exacerbation. In the latter case both ivstart
** and ivstop dates are missing
insheet id site trt rand lastfu fev ivstart ivstop using "dnase.csv", comma
      ** below is an explanation of the variables in the dataset
      * id: subject id
      * site: institution id
      * trt: (treatment arm) 0=placebo, 1=rhDNase
      * rand: randomization date
      * lastfu: last follow-up date on study
      * fev: (forced expiratory volume), a measure of lung capacity
      * ivstart: start of infection/exacerbation (if non-missing)
      * ivstop: end of infection/exacerbation (if non-missing)

      label define trt 0 "Placebo" 1 "rhDNase"
      label values trt trt

* replace FEV with a version rescaled by dividing by ten to "make
* coefficients be about the same size" in the words of Therneau & Grambsch
      replace fev = round(fev/10,0.01)

* generate Stata date variables from the text date variables in the rhDNase
* dataset
      gen rand_dt = date(rand, "DMY")
      gen lastfu_dt = date(lastfu, "DMY")
      gen start_dt = date(ivstart, "DMY")
      gen stop_dt = date(ivstop, "DMY")
      format rand_dt lastfu_dt start_dt stop_dt %dD_m_Y

* drop the rand, lastfu, ivstart and ivstop text variables, we now have Stata
* date versions
      drop rand lastfu ivstart ivstop
* generate futime and new ivstart/ivstop variables in "days since
* randomization" form
      gen ivstart = start_dt - rand_dt
      gen ivstop = stop_dt - rand_dt
      gen futime = lastfu_dt - rand_dt

* drop the Stata date variables (and site) as we only need the "days from
* randomization" variables
      drop *_dt site
* generate an event indicator to be 1 if the record corresponds to an
* exacerbation, 0 otherwise
      gen status = 1
      replace status = 0 if ivstart == . & ivstop == .

* save the dataset in its current format before manipulating it for the Cox
```

```
* models. This dataset is now equivalent to the rhDNase dataset provided with
* the R 'tpr' package except for the centering of FEV (it is centered in the
* tpr version and not here).
      save "Therneau_Grambsch_rhDNase.dta", replace

* create a set of additional records for subjects who had at least one
* exacerbation, this record will track the time from the end of their final
* exacerbation until end of the study (if it exists)

* drop records that correspond to exacerbations that started before
* randomization (because we don't want those to count for someone having an
* exacerbation)
           drop if ivstart < 0 & ivstop == futime

* drop all non-exacerbation records (this drops all subjects without an
* exacerbation)
      drop if status == 0

* sort records for all subjects with an exacerbation by exacerbation end day
* and keep only their final record, set ivstart and ivstop to missing and
* status to 0 to make this record the same format as the records for subjects
* without an exacerbation
      sort id ivstart
      bysort id: keep if _n == _N
      replace ivstart = .
      replace ivstop = .
      replace status = 0

* merge these additional records with the dataset that was saved earlier (and
* sort by event start)
      append using "U:\ArthurIndependentStudy\Therneau_Grambsch_rhDNase.dta"
      sort id ivstart


* Restructure the dataset with new variables time1, time2 and enum - these
variables will represent:
* time1 - the start of an "at risk" interval for a subject, this is 0 for
*         their first record and the end of the previous exacerbation for any
*         subsequent records
* time2 - the end time of an "at risk" interval for a subject, this is either
          the start of an exacerbation or the end of follow up
* enum - a counter to keep track of what exacerbation a subject is "at risk"
*        for, this is 1 for first records, 2 for second records, and so on
* generate the enum variable
      bysort id: gen enum = _n
* generate an initial time1 set to 0 (this is only correct for first records)
      gen time1 = 0
* generate an initial time2 set to the start of the exacerbation each record
* represents (this is correct for all but subjects with no events and the
* final record for each subject)
      gen time2 = ivstart
* if time2 is missing the record is one of the above exceptions, so it should
* be end of follow up
      replace time2 = futime if time2 == .
* replace time1 with the end of a subject's previous exacerbation for all but
* their first record
      bysort id: replace time1 = ivstop[_n-1] if _n != 1
```

```
* add 6 days to the end of exacerbation lengths, this is to account for the
* rhDNase trial exacerbation definition which requires 6 days of exacerbation
* free time after an event before being "at risk" for another - this is done
* by adding 6 days to time1 for a subject's records succeeding exacerbations
      bysort id: replace time1 = time1 + 6 if time1 != 0 & status[_n-1] == 1

* drop records associated with pre-randomization exacerbations, these exist
* for subjects 173, 432, 436 and 450 and decrease enum by one for all records
* from those subjects (since the pre-randomization exacerbation was counted
* as an exacerbation previously, but should not be)
      drop if ivstart < 0
      replace enum = enum - 1 if id == 173 | id == 432 | id == 436 | id == ///
      450

* Alter entrance times for the subjects with pre-randomization events the new
entrance time includes the 6 day post-exacerbation exacerbation-free period
      replace time1 = 13 if enum == 1 & id == 173
      replace time1 = 9 if enum == 1 & id == 432
      replace time1 = 37 if enum == 1 & id == 436
      replace time1 = 11 if enum == 1 & id == 450

* drop ivstart and ivstop variables (we won't need them going forward)
      drop ivstart ivstop

* drop subjects' final records if they don't account for any time on study
* "at risk"
      drop if time1 >= time2 & time1 >= futime

* save this dataset under the same name, as this is the final dataset we need
* for application of the AG and Conditional extensions of the Cox model (and
* we no longer need the previous dataset
      save "Therneau_Grambsch_rhDNase.dta", replace


*** Checking Exacerbation count results in Table 8.12 of the text (these
*** numbers match)
      bysort id: egen num_exacerbations = total(status)
      tab trt num_exacerbations if enum == 1
      drop num_exacerbations


*** Below are the "First event", "Andersen-Gill", and two "Conditional"
*** analyses from Table 8.13

* Time to first analysis (time2 is our exacerbation (event) time, time1 is
* enrollment time and restricting to enum == 1 selects only the first records
* for all subjects)
      stset time2 if enum == 1, failure(status) time0(time1)
      stcox trt fev, nolog nohr efron
      stcox trt fev, nolog nohr efron robust


* Andersen-Gill
      stset time2, failure(status) time0(time1)
      stcox trt fev, cluster(id) nolog nohr efron
      stcox trt fev, nolog nohr efron robust
```

```
* Conditional
      stcox trt fev, strata(enum) nolog nohr efron
      stcox trt fev, cluster(id) strata(enum) nolog nohr efron

      ** generating a collapsed enum variable for the 3 strata models
      recode enum (1=1) (2=2) (3/5 = 3), generate(enum_modified)

* Conditional/3
      stcox trt fev, strata(enum_modified) nolog nohr efron
      stcox trt fev, cluster(id) strata(enum_modified) nolog nohr efron


* Conditional/3 with treatment and fev interaction based on strata analysis
* from Table 8.14
      stcox c.fev#i.enum_modified c.trt#i.enum_modified, ///
      strata(enum_modified) nolog nohr efron
      stcox c.fev#i.enum_modified c.trt#i.enum_modified, cluster(id) ///
      strata(enum_modified) nolog nohr efron


* Data manipulation to create the appropriate risk sets for application of
* the WLW (Marginal) model

* set up records for time "at risk" for a first exacerbation - this case is
* straight forward as it simply requires keeping all of the records for the
* first enum level
      clear
      use "Therneau_Grambsch_rhDNase.dta"
      keep if enum == 1
      save "temp1.dta", replace

* set up records for time "at risk" for a second exacerbation - keep all
* records for the first or second strata, any first exacerbation strata
* record ending in an event needs it's exacerbation status set to 0 since it
* is not a second exacerbation, lastly, the records we are keeping from the
* first enum strata must be relabeled since we want them to be attributed to
* the second  strata
      clear
      use "Therneau_Grambsch_rhDNase.dta"
      keep if enum < 3
      replace status = 0 if enum < 2
      replace enum = 2
      save "temp2.dta", replace


* set up records for time "at risk" for a third exacerbation - keep all
* records for the first, second or third strata, any first or second strata
* record ending in an event needs it's exacerbation status set to 0 since it
* is not a third exacerbation, lastly, the records we are keeping from the
* first two strata must be relabeled since we want them to be attributed to
* the third strata
      clear
      use "Therneau_Grambsch_rhDNase.dta"
      keep if enum < 4
      replace status = 0 if enum < 3
      replace enum = 3
```

```
      save "temp3.dta", replace

* set up records for time "at risk" for a fourth exacerbation - same process
* as above
      clear
      use "Therneau_Grambsch_rhDNase.dta"
      keep if enum < 5
      replace status = 0 if enum < 4
      replace enum = 4
      save "temp4.dta", replace

* set up records for time "at risk" for a fifth exacerbation - same process
* as above
      clear
      use "Therneau_Grambsch_rhDNase.dta"
      replace status = 0 if enum < 5
      replace enum = 5
      save "temp5.dta", replace

* merge all of the different exacerbation strata datasets that were just
* created
      clear
      use "temp1.dta"
      append using "temp2.dta"
      append using "temp3.dta"
      append using "temp4.dta"
      append using "temp5.dta"
      save "Therneau_Grambsch_rhDNase_Marginal.dta", replace

*** Below are the "WLW" (Marginal) analyses from Table 8.13 and Table 8.14

* WLW (Marginal)
      stset time2, failure(status) time0(time1)
      stcox trt fev, strata(enum) nolog nohr efron
      stcox trt fev, cluster(id) strata(enum) nolog nohr efron

      ** generating a collapsed enum variable
      recode enum (1=1) (2=2) (3/5 = 3), generate(enum_modified)

* WLW/3 (Marginal with 3 strata)
      stcox trt fev, cluster(id) strata(enum_modified) nolog nohr efron
      stcox trt fev, strata(enum_modified) nolog nohr efron



* WLW/3 with treatment and fev interaction based on strata
      stcox c.fev#i.enum_modified c.trt#i.enum_modified, ///
      strata(enum_modified) nolog nohr efron
      stcox c.fev#i.enum_modified c.trt#i.enum_modified, cluster(id) ///
      strata(enum_modified) nolog nohr efron
```

10.5. **Yan and Fine R Verification Code:**

```
## R code to verify the results presented on page 507 of Yan & Fine (Analysis
## of Episodic Data with Application to Recurrent Pulmonary Exacerbations in
## Cystic Fibrosis Patients, 2008)

## load the TPR package and the included rhDNase dataset, dnase
      library(tpr)
      data(dnase)

## undo the centering of the rescaled FEV (mean obtained from Therneau &
## Grambasch dataset) see appendix notes for an explanation of (and
## motivation for) this step
      dnase$fev <- dnase$fev + 59.51082/10

## remove records for episodes starting before baseline (pre-randomization
## exacerbations) see appendix notes for an explanation of (and motivation
## for) this step as well
      dnase <- dnase[dnase$iv1 > 0 | is.na(dnase$iv1),]



## construct temporal process response for total number of exacerbations


## before calling the TPR function itself we need two things:
## 1) a response process
## 2) a time-independent covariate matrix

## step 1) creating a response process consisting of a list of lgtdl objects
## (one for each unique id)

## extract the unique id and subject level covariates
      dat <- unique(dnase[,c("id", "futime", "fev", "rx")])

## this rec object is a list of list, and will be used to construct our
## response process yrec, each sublist in rec is named with the id of a
## unique subject and contains a vector with their follow up start day,
## exacerbation episode start day(s) if the subject had any, and the end if
## their follow up. For example, rec[[21]] returns a vector of (0, 20, 62,
## 169) which tells us subject 21 started follow up at day 0, had
## exacerbation episodes start at days 20 and 62 and ended follow up at day
## 169
      rec <- lapply(split(dnase[,c("id", "iv1", "futime")], dnase$id),
            function(x) {
                        v <- x$iv1
                        maxfu <- max(x$futime)
                        if (is.na(v[1])) c(0, maxfu)
                        else c(0, v[!is.na(v)], maxfu)
                        })


## the yrec object is our response process and is a list of lgtdl objects,
## each sublist is named with the id of a unique subject and contains an
## lgtdl object with time and one covariate, the time contains their follow
## up start day, exacerbation start day(s), and their follow up stop day, the
```

```
## covariate "cov" contains the exacerbation episode number that the ith
## entry represents (the last entry is just the end of follow up, so cov does
## not increase) for example, yrec[[21]] returns a data frame/lgtdl object
## with 4 rows with time values of: (0, 20, 62, 169)  and covariate values of
## (0, 1, 2, 2) which tells us that subject 21 started follow up at day 0,
## had their first exacerbation episode at day 20, had their second at day
## 62, and stopped follow up at day 169
      yrec <- lapply(rec,
            function(x) {
                        dat <- data.frame(time=x, cov=1:length(x)-1)
                        len <- length(x)
                        dat$cov[len] <- dat$cov[len - 1]
                        as.lgtdl(dat)
                        })


## In addition to the response process we need a data availability indicator
process:

## rt contains a list of lists, each sublist is named with the id of a unique
## subject and contains an lgtdl object with time and one covariate. This
## object indicates when each subject is under
## observation, so this time starts at zero, has an entry for when the
## subject's follow up ended and has a final entry for the maximal follow up
## of the dataset. The covariate "cov" indicates the subject's "at risk"
## status, so it starts at 1, and is changed to zero at the end of their
## follow up
      tu <- max(dat$futime)
      rt <- lapply(1:nrow(dat),
                  function(i) {
                        x <- dat[i, "futime"]
                        time <- c(0, x, tu)
                        cov <- c(1, 0, 0)
                        as.lgtdl(data.frame(time=time, cov=cov))
                        })


## step 2) time-independent covariate matrix for treatment and baseline FEV
      xmat <- model.matrix(~ rx + fev, data=dat)


## TPR function call for the total number of exacerbations process to create
## the TPR object m.rec, below is a brief description of the different
## arguments passed to the TPR function

## yrec is the response "y", as indicated above, it a list of lists
   # containing an lgtdl object for each unique subject, each object contains
   # the observation start day, the subject's exacerbation episode
   # start days and their follow up stop day as well as a counter of how many
   # exacerbation episodes they've had at time t.
## rt is the "delta" object indicating "data availability" or "at risk"
   # status of subjects (when their follow up started and stopped, plus an
   # entry for when the maximal follow up ended)
## xmat is the "x" object, a "covariate matrix for time-varying coefficients"
   # for fixed covariates (in this case treatment and baseline FEV)
## "xtv" specifies a "list of time-varying covariates with time-varying
```

```
   # coefficients", in this case list() indicates there are none
## "z" specifies the "covariate matrix for time-independent coefficients",
   # requires a matrix of size n by 0 where n is the number of unique
   # subjects when not being used, documentation indicates this feature is
   # not ready/implemented yet and it is not used in these analyses
## "tis" specifies "the time points at which the model is to be fit", in this
   # case from days 10 to 168 in daily increments
## "w" specifies the "weight vector", in this case, we use even weighting
## "family" specifies the response distribution, Poisson is used here for the
   # exacerbation counts
## "evstr" specifies the link and variance functions, in this case (link = 5,
   # v = 3) specifies a log link and Poisson variance

      tis <- 10:168
      m.rec <- tpr(y=yrec,delta=rt,x=xmat[,1:3],xtv=list(),
            z=xmat[,-(1:3),drop=FALSE],
            tis=10:168, w = rep(1, length(tis)), family = poisson(),
            evstr = list(link = 5, v = 3))


## plots of the time-varying coefficients from the total number of
## exacerbations TPR fit - these unexponentiated plots correspond to those in
## Figure 4(a) m.rec$alpha contains the time varying coefficients for the
## intercept and fixed covariates: treatment group, and baseline FEV. These
## are unexponentiated coefficients, add options "fun=poisson()$linkinv" and
## "fun = poisson()$mu.eta" for exponentiated coefficient plots
      par(mfrow=c(1,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,0,0))
      for(i in 1:3) ci.plot(m.rec$tis, m.rec$alpha[,i],
      sqrt(m.rec$valpha[,i]))


      for(i in 1:3) {ci.plot(m.rec$tis, m.rec$alpha[,i],
      sqrt(m.rec$valpha[,i]),fun=poisson()$linkinv,
                  dfun = poisson()$mu.eta)
                  if (i> 1) abline(h = 1, v = 0, col = "gray60",lty =2)}


## integral test statistic in Table 2 for Model (a) (note: these do not match
## as discussed in the appendix notes)
      sig.test.int.ff(m.rec, idx=2:3,weight = TRUE, ncut=0)


## construct temporal process response for accumulative days in exacerbation

## do1.acc is a function used to construct the yacc response below
      do1.acc <- function(x) {
            gap <- x$iv2 - x$iv1 + 1
            if (all(is.na(gap))) yy <- tt <- NULL
            else {
            gap <- na.omit(gap)
            yy <- cumsum(rep(1, sum(gap)))
            tt <- unlist(sapply(1:length(gap), function(i) seq(x$iv1[i],
                  x$iv2[i], by=1.0)))
            }
      yy <- c(0, yy, rev(yy)[1])
      tt <- c(0, tt, max(x$futime))
```

```
      as.lgtdl(data.frame(time=tt, cov=yy))
      }

## the yacc object is a list of lists, each sublist is named with the id of a
## unique subject and contains a lgtdl object with time and one covariate,
## the time contains the days in which that subject is in an exacerbation
## episode and the covariate is the total number of days spent in
## exacerbation episodes up to that time
      yacc <- lapply(split(dnase[,c("id", "iv1", "iv2", "futime")],
            dnase$id), do1.acc)

## time-independent covariate matrix for treatment and baseline FEV
      xmat <- model.matrix(~ rx + fev, data=dat)


## TPR function call for the accumulated days in exacerbation process,
## creates the TPR object m.acc
      tis = 10:168
      m.acc <- tpr(yacc, rt, xmat[,1:3], list(), xmat[,-(1:3),drop=FALSE],
            list(),tis=tis, w = rep(1, length(tis)), family = gaussian(),
            evstr = list(link = 1, v = 1))

## plots of the time-varying coefficients from the accumulated days in
## exacerbation TPR fit these plots correspond to those in Figure 4(b)
      par(mfrow=c(1,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,0,0))
      for(i in 1:3) ci.plot(m.acc$tis, m.acc$alpha[,i],
            sqrt(m.acc$valpha[,i]))

## integral test statistic in Table 2 for Model (b) (note: these do not match
## as discussed in the appendix notes)
      sig.test.int.ff(m.acc, idx=2:3,weight = TRUE, ncut=0)


## construct temporal process response for proportion of days in exacerbation

## do1.prop is a function used to construct the yprop response below
      do1.prop <- function(x) {
            gap <- x$iv2 - x$iv1 + 1
            if (all(is.na(gap))) fu <- y.prop <- NULL
            else {
                  gap <- na.omit(gap)
                  tt <- unlist(sapply(1:length(gap), function(i)
                        seq(x$iv1[i], x$iv2[i], by=1.0)))
                  fu <- seq(1,x$futime[1])
                  resp <- rep(0,length(fu))
                  resp[tt] = 1
                  y.prop  <- cumsum(resp)/fu
                  }
            y.prop <- c(0, y.prop, rev(y.prop)[1])
            fu <- c(0, fu, max(x$futime) + 1.0)
            as.lgtdl(data.frame(time=fu, cov=y.prop))
            }

## yprop is also a list of lists, each sublist is named with the id of a
## unique subject and contains a lgtdl object with time and one covariate,
## the time contains the days for which that subject is under observation and
```

```
## the covariate contains the proportion of days that subject has spent in
## exacerbation up to time t
      yprop <- lapply(split(dnase[,c("id", "iv1", "iv2", "futime")],
            dnase$id), do1.prop)

## time-independent covariate matrix for treatment and baseline FEV
      xmat <- model.matrix(~ rx + fev, data=dat)

## TPR function call for the proportion of days in exacerbation process,
## creates the TPR object m.prop
      tis <- 10:168
      m.prop <- tpr(yprop, rt, xmat[,1:3], list(), xmat[,-(1:3),drop=FALSE],
            list(),
            tis=tis, w = rep(1, length(tis)), binomial(link = "logit"),
            evstr = list(link = 2, v = 2))

## plots of the time-varying coefficients from the proportion of days in
## exacerbation TPR fit these plots correspond to those in Figure 4(c)
      par(mfrow=c(1,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,0,0))
      for(i in 1:3) ci.plot(m.prop$tis, m.prop$alpha[,i],
      sqrt(m.prop$valpha[,i]))

## integral test statistic in Table 2 for Model (c) (note: these do not match
## as discussed in the appendix notes)
      sig.test.int.ff(m.prop, idx=2:3,weight = TRUE, ncut=0)
```

11. Appendix C - ISIS Trial Analyses Code:

11.1. Stata Code for Extensions of the Cox Model:

```
** Stata code for the application of AG/Conditional and Marginal extensions
** of the Cox model. These datasets are not publically available at this time

* read in the prepared ISIS dataset in the AG/Conditional format, this
* dataset is in the same form as the rhDNase dataset constructed to verify
* Therneau and Grambsch's analyses. Key variables represent:

* id - participant identification number
* trt - treatment arm (0 = Isotonic Saline (control), 1 = Hypertonic Saline)
* pft - indicates whether or not a subject was a participant in the iPFT
*      substudy (0 = No, 1 = Yes)
* sex - participant sex (0 = Male, 1 = Female)
* age_years - participant age at enrollment (in years)
* ht_cm - participant height at enrollment (in cm)
* wt_kg - participant weight at enrollment (in kg)
* futime - participant days of observation time (follow up in days from
*     randomization)
* fev_scaled - participant FEV at enrollment if subject was a part of the
      iPFT substudy and had an acceptable baseline measurement (missing
      otherwise). These measurements have been divided by the standard
      deviation of enrollment measures of FEV 0.5
* status - indicates if the record is associated with an exacerbation event
*     (0 = No, 1 = Yes)
* time1 - the start day of an "at risk" interval for a subject, this is 0 for
*     first records and the end of the previous exacerbation for any
*     subsequent records
* time2 - the end time of an "at risk" interval for a subject, this is either
*     the start of an exacerbation or the end of follow up
* enum - a counter to keep track of what exacerbation a subject is "at risk"
*     for, this is 1 for first records, 2 for second records, and so on
      use "ISIS AG-conditional format.dta"

* tabulate exacerbation counts
      bysort id: egen num_exacerbations = total(status)
      tab trt num_exacerbations if enum == 1
      drop num_exacerbations


*** below are the First Event, Andersen-Gill, and three Conditional analyses

* time to first exacerbation model
      stset time2 if enum == 1, failure(status) time0(time1)
      stcox trt, nolog robust

* plot of Kaplan-Meier estimates by treatment arm for all subjects
      sts graph, by(trt) risktable(0 84 168 252 336 420, ///
      order(1 "Isotonic Saline" 2 "Hypertonic Saline")) ///
      legend(label(1 "Isotonic Saline") label(2 "Hypertonic Saline")) ///
      xtitle("Days since randomization") ///
      xlabel(0(84)420) title("Kaplan-Meier survival estimates - All ///
      participants", size(medsmall)) plot2opts(lpattern(dash))
```

```
* Andersen-Gill model
      stset time2, failure(status) time0(time1)
      stcox trt, cluster(id) nolog

* Conditional model
      stcox trt, cluster(id) strata(enum) nolog

* generate collapsed enum variables, one for 3 strata as used in Therneau &
* Grambsch's rhDNase analysis and one with 5 strata
      recode enum (1=1) (2=2) (3=3) (4=4) (5/10=5), generate(enum_5)
      recode enum (1=1) (2=2) (3/10=3), generate(enum_3)


* Conditional - 3 strata model
      stcox trt, cluster(id) strata(enum_3) nolog

* Conditional - 5 strata model
      stcox trt, cluster(id) strata(enum_5) nolog


*** below are the Conditional analyses with 3 and 5 strata treatment
*** interactions

* Conditional - 3 strata with treatment interaction based on strata
      stcox c.trt#i.enum_3, cluster(id) strata(enum_3) nolog

* Conditional - 5 strata with treatment interaction based on strata
      stcox c.trt#i.enum_5, cluster(id) strata(enum_5) nolog


*** below are the AG and Conditional analyses for iPFT substudy participants

* time to first exacerbation model - iPFT subjects with acceptable enrollment
* FEV
      stset time2 if enum == 1& pft == 1, failure(status) time0(time1)
      stcox trt fev_scaled age_years ht_cm sex, robust

* plot of Kaplan-Meier estimates by treatment arm for iPFT subjects
      sts graph if pft == 1, by(trt) risktable(0 84 168 252 336 420, ///
      order(1 "Isotonic Saline" 2 "Hypertonic Saline")) ///
      legend(label(1 "Isotonic Saline") label(2 "Hypertonic Saline")) ///
      xtitle("Days since randomization") ///
      xlabel(0(84)420) title("Kaplan-Meier survival estimates - iPFT ///
      participants", size(medsmall)) plot2opts(lpattern(dash))

* Andersen-Gill   model - iPFT subjects with acceptable enrollment FEV
      stset time2 if pft == 1, failure(status) time0(time1)
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) nolog

* Conditional - iPFT subjects with acceptable enrollment FEV
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) strata(enum) ///
      nolog

* Conditional - 3 strata model - iPFT subjects with acceptable enrollment FEV
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) ///
```

```
      strata(enum_3) nolog

* Conditional - 5 strata model - iPFT subjects with acceptable enrollment FEV
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) ///
      strata(enum_5) nolog


*** below is the Conditional analysis with 3 strata treatment interaction for
*** iPFT substudy participants

* Conditional - 3 strata with treatment interaction - iPFT subjects with
* acceptable enrollment FEV
      stcox c.trt#i.enum_3 c.fev_scaled#i.enum_3 age_years ht_cm sex, ///
      cluster(id) strata(enum_3) nolog


*** Data management to create the appropriate risk sets for application of
the WLW Marginal model

* keep all of the records for first at risk strata, this case is fairly
straightforward
      clear
      use "ISIS AG-conditional format.dta"
      keep if enum == 1
      save "temp1.dta", replace

* set up records for the second at risk strata, we want to keep all records
* for the first or second strata and any first strata record ending in an
* event needs it's status reset since it is not an event for the second
* strata, lastly, the records we are keeping from the first strata must be
* relabeled since we want them to be attributed to strata 2
      clear
      use "ISIS AG-conditional format.dta"
      keep if enum < 3
      replace status = 0 if enum < 2
      replace enum = 2
      save "temp2.dta", replace

* set up records for the third at risk strata, we want to keep all records
* for the first through third strata and any first/second strata records
* ending in an event needs their status reset since it is not an event for
* strata 3, lastly, the records we are keeping from strata 1/2 must be
* relabeled since we want them to be attributed to strata 3
      clear
      use "ISIS AG-conditional format.dta"
      keep if enum < 4
      replace status = 0 if enum < 3
      replace enum = 3
      save "temp3.dta", replace

* analogous process for the fifth strata
      clear
      use "ISIS AG-conditional format.dta"
      keep if enum < 5
      replace status = 0 if enum < 4
      replace enum = 4
```

```
      save "temp4.dta", replace

* analogous process for the sixth strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 5
      replace enum = 5
      save "temp5.dta", replace

* analogous process for the seventh strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 6
      replace enum = 6
      save "temp6.dta", replace

* analogous process for the eighth strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 7
      replace enum = 7
      save "temp7.dta", replace

* analogous process for the ninth strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 8
      replace enum = 8
      save "temp8.dta", replace

* analogous process for the tenth strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 9
      replace enum = 9
      save "temp9.dta", replace

* analogous process for the eleventh strata
      clear
      use "ISIS AG-conditional format.dta"
      replace status = 0 if enum < 10
      replace enum = 10
      save "temp10.dta", replace

* append all the strata specific datasets that were just created
      clear
      use "temp1.dta"
      append using "temp2.dta"
      append using "temp3.dta"
      append using "temp4.dta"
      append using "temp6.dta"
      append using "temp7.dta"
      append using "temp8.dta"
      append using "temp9.dta"
      append using "temp10.dta"
      save "ISIS WLW-marginal format.dta"
```

```
*** below are the three Marginal model analyses

* Marginal model
      stset time2, failure(status) time0(time1)
      stcox trt, cluster(id) strata(enum) nolog

* generate a collapsed enum variable
      recode enum (1=1) (2=2) (3=3) (4=4) (5/10=5), generate(enum_5)
      recode enum (1=1) (2=2) (3/10=3), generate(enum_3)

* Marginal - 3 strata model
      stcox trt, cluster(id) strata(enum_3) nolog

* Marginal - 5 strata model
      stcox trt, cluster(id) strata(enum_5) nolog


*** below are the Marginal analyses with 3 and 5 strata treatment
*** interactions

* Marginal - 3 strata with treatment interaction based on strata
      stcox  c.trt#i.enum_3, cluster(id) strata(enum_3) nolog

* Marginal - 5 strata with treatment interaction based on strata
      stcox  c.trt#i.enum_5, cluster(id) strata(enum_5) nolog


*** below are the Marginal analyses for iPFT substudy participants

* Marginal model - iPFT subjects with acceptable enrollment FEV
      stset time2 if pft==1, failure(status) time0(time1)
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) ///
      strata(enum) nolog

* Marginal - 3 strata model - iPFT subjects with acceptable enrollment FEV
      stcox trt fev_scaled age_years ht_cm sex, cluster(id) ///
      strata(enum_3) nolog


*** below is the Marginal analysis with 3 strata treatment interaction for
*** iPFT substudy participants

* Marginal - 3 strata with treatment interaction - iPFT subjects with
* acceptable enrollment FEV
      stcox  c.trt#i.enum_3 c.fev_scaled#i.enum_3 age_years ht_cm sex,
      cluster(id) strata(enum_3) nolog
```

11.2. **R Code for Temporal Process Regression (all participants):**

```
## R Code for Temporal Process Regression analyses of the ISIS Trial
## exacerbation data. Refer to the "Yan and Fine R Verification Code" for
## additional comments regarding set up and application of the 'tpr' package
## load the TPR package and the foreign package (to read in Stata .dta files)
      library(tpr)
      library(foreign)

## read in the ISIS data
      isis <-read.dta("ISIS TPR format.dta")

## rename variables from the ISIS dataset
      names(isis)[names(isis) == "trt"] <- "rx"
      names(isis)[names(isis) == "start"] <- "iv1"
      names(isis)[names(isis) == "stop"] <- "iv2"

## replace rx, which is currently a factor variable, with a 0/1 indicator
## (1=HS,0=IS)
      isis$rx <- (as.numeric(isis$rx)-1)

## construct temporal process response for total number of exacerbations

## extract the unique id and subject level covariates
      dat <- unique(isis[,c("id", "futime", "rx")])


      rec <- lapply(split(isis[,c("id", "iv1", "futime")], isis$id),
            function(x) {
                  v <- x$iv1
                  maxfu <- max(x$futime)
                  if (is.na(v[1])) c(0, maxfu)
                  else c(0, v[!is.na(v)], maxfu)
                  })

      yrec <- lapply(rec,
            function(x) {
                  dat <- data.frame(time=x, cov=1:length(x)-1)
                  len <- length(x)
                  dat$cov[len] <- dat$cov[len - 1]
                  as.lgtdl(dat)
                  })

## construct data availability indicator process
      tu <- max(dat$futime)
      rt <- lapply(1:nrow(dat),
            function(i) {
                  x <- dat[i, "futime"]
                  time <- c(0, x, tu)
                  cov <- c(1, 0, 0)
                  as.lgtdl(data.frame(time=time, cov=cov))
                  })

## construct time-independent covariate matrix for treatment and baseline FEV
      xmat <- model.matrix(~ rx, data=dat)
```

```
## TPR function call for the total number of exacerbations process to create
## the TPR object m.rec the "tis" object here is specified as days 10 through
## 357, we begin estimating coefficients at day 10 due to large variability
## before that time (day 10 was also used in the rhDNase analyses) and we
## stop at day 357 as this corresponds to the latest day from randomization
## at which a final visit was to occur (48 +/- 3 weeks was the specified
## final visit window, 51*7 = 357 days)
      tis <- 10:357
      m.rec <- tpr(y=yrec,delta=rt,x=xmat[,1:2],xtv=list(),
      z=xmat[,-(1:2),drop=FALSE],
      tis=tis, w = rep(1, length(tis)), family = poisson(),
      evstr = list(link = 5, v = 3))

## plots of the time-varying coefficients from the total number of
exacerbations TPR fit
      par(mfrow=c(1,2), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      colnames(xmat) <- c("Intercept coefficient over time for IS arm",
            "Treatment coefficient over time comparing HS to IS")
            for(i in 1:2) {ci.plot(m.rec$tis, m.rec$alpha[,i],
                  sqrt(m.rec$valpha[,i]),xlab=colnames(xmat)[i])
                  if (i> 1) abline(h = 0, col = "gray60",lty =2)}
      title("Number of Exacerbations - All ISIS Subjects - unexponentiated
      coefficients", outer=TRUE)

## plots of the exponentiated coefficients
      par(mfrow=c(1,2), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      colnames(xmat) <- c("Mean number of exacerbations for IS arm",
            "Ratio of mean number of exacerbations comparing HS to IS")
      for(i in 1:2) {ci.plot(m.rec$tis, m.rec$alpha[,i],
            sqrt(m.rec$valpha[,i]),fun=exp,dfun=exp,
            xlab=colnames(xmat)[i])
            if (i> 1) abline(h = 1, col = "gray60",lty =2)}
      title("Number of Exacerbations - All ISIS Subjects", outer=TRUE)


## construct temporal process response for accumulative days in exacerbation
      do1.acc <- function(x) {
      gap <- x$iv2 - x$iv1 + 1
      if (all(is.na(gap))) yy <- tt <- NULL
      else {
            gap <- na.omit(gap)
            yy <- cumsum(rep(1, sum(gap)))
            tt <- unlist(sapply(1:length(gap), function(i) seq(x$iv1[i],
                  x$iv2[i], by=1.0)))
            }
      yy <- c(0, yy, rev(yy)[1])
      tt <- c(0, tt, max(x$futime))
      as.lgtdl(data.frame(time=tt, cov=yy))
      }


      yacc <- lapply(split(isis[,c("id", "iv1", "iv2", "futime")],
             isis$id),do1.acc)
```

```
## TPR function call for the accumulated days in exacerbation process,
creates the TPR object m.acc
      tis <- 10:357
      m.acc <- tpr(yacc, rt, xmat[,1:2], list(), xmat[,-(1:2),drop=FALSE],
      list(),tis=tis, w = rep(1, length(tis)), family = gaussian(),
      evstr = list(link = 1, v = 1))


## plots of the time-varying coefficients from the accumulated days in
## exacerbation TPR fit
      par(mfrow=c(1,2), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      colnames(xmat) <- c("Mean days in exacerbation for IS arm",
            "Difference in mean days in exacerbation comparing HS to IS")
      for(i in 1:2) ci.plot(m.acc$tis, m.acc$alpha[,i],
            sqrt(m.acc$valpha[,i]),xlab=colnames(xmat)[i])
      title("Accumulated Days in Exacerbation - All ISIS Subjects",
      outer=TRUE)
```

11.3. R Code for Temporal Process Regression (iPFT participants):

```
## R Code for Temporal Process Regression analyses of the ISIS Trial iPFT
## substudy exacerbation data. Refer to the "Yan and Fine R Verification
## Code" for additional comments regarding set up and application of the
'tpr' package

## load the TPR package and the foreign package (for reading in of Stata .dta
files)
      library(foreign)
      library(tpr)

## read in the ISIS data and keep only the iPFT substudy participants
      isis.pft <-read.dta("ISIS TPR format.dta")
      isis.pft <- isis.pft[!is.na(isis.pft$fev),]

## rename variables from the ISIS dataset
      names(isis.pft)[names(isis.pft) == "trt"] <- "rx"
      names(isis.pft)[names(isis.pft) == "start"] <- "iv1"
      names(isis.pft)[names(isis.pft) == "stop"] <- "iv2"

      names(isis.pft)[names(isis.pft) == "fev"] <- "fev.unscaled"
      names(isis.pft)[names(isis.pft) == "fev_scaled"] <- "fev"
      names(isis.pft)[names(isis.pft) == "age_years"] <- "age"
      names(isis.pft)[names(isis.pft) == "ht_cm"] <- "height"

## replace rx, which is currently a factor variable, with a 0/1 indicator
## (1=HS, 0=IS)
      isis.pft$rx <- (as.numeric(isis.pft$rx)-1)

## replace sex, which is currently a factor variable, with a 0/1 indicator
## (1=HS, 0=IS)
      isis.pft$sex <- (as.numeric(isis.pft$sex)-1)

## replace the SD scaled FEV with a centered version so the intercept
coefficient estimate is for a
## subject with mean baseline FEV - not zero baseline FEV
      isis.pft$fev <- isis.pft$fev - mean(isis.pft$fev)

## replace age with a centered version (that is also on the month scale) so
the intercept coefficient
## estimate is for a subject with mean baseline age - not zero baseline age
      isis.pft$age <- (isis.pft$age - mean(isis.pft$age))*12

## replace height with a centered version so the intercept coefficient
estimate is for a subject with
## mean baseline height - not zero baseline height
      isis.pft$height <- isis.pft$height - mean(isis.pft$height)


## construct temporal process response for recurrent event

## extract the unique id and subject level covariates
      dat <- unique(isis.pft[,c("id", "futime", "fev","age", "height", "sex",
      "rx")])
```

```
      rec <- lapply(split(isis.pft[,c("id", "iv1", "futime")], isis.pft$id),
            function(x) {
                  v <- x$iv1
                  maxfu <- max(x$futime)
                  if (is.na(v[1])) c(0, maxfu)
                  else c(0, v[!is.na(v)], maxfu)
            })

      yrec <- lapply(rec,
            function(x) {
                  dat <- data.frame(time=x, cov=1:length(x)-1)
                  len <- length(x)
                  dat$cov[len] <- dat$cov[len - 1]
                  as.lgtdl(dat)
            })

## construct data availability (or at risk) indicator process
      tu <- max(dat$futime)
      rt <- lapply(1:nrow(dat),
            function(i) {
                  x <- dat[i, "futime"]
                  time <- c(0, x, tu)
                  cov <- c(1, 0, 0)
                  as.lgtdl(data.frame(time=time, cov=cov))
            })

## construct time-independent covariate matrix for treatment and baseline
## FEV, height, age and sex
      xmat <- model.matrix(~ rx + fev + age + height + sex, data=dat)


## TPR function call for the total number of exacerbations process to create
## the TPR object m.rec the "tis" object here is specified as days 10 through
## 357, we begin estimating coefficients at day 10 due to large variability
## before that time (day 10 was also used in the rhDNase analyses) and we
## stop at day 357 as this corresponds to the latest day from randomization
## at which a final visit was to occur (48 +/- 3 weeks was the specified
## final visit window, 51*7 = 357 days)
      tis <- 10:357
      m.rec <- tpr(y=yrec,delta=rt,x=xmat[,1:ncol(xmat)],xtv=list(),z=xmat[,-
(1:ncol(xmat)),drop=FALSE],
      tis=tis, w = rep(1, length(tis)), family = poisson(),
      evstr = list(link = 5, v = 3))


## plots of the time-varying coefficients from the total number of
exacerbations TPR iPFT fit
      par(mfrow=c(2,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      colnames(xmat) <- c("Intercept coefficient over time for males in IS
      arm with mean FEV 0.5, age,
      height at baseline","Treatment coefficient over time comparing HS to
      IS","FEV 0.5 coefficient over time for one SD difference",
      "Age coefficient over time for one month difference",
      "Height coefficient over time for one cm difference",
      "Sex coefficient over time comparing females to males")
```

```
      for(i in 1:ncol(xmat)) {ci.plot(m.rec$tis, m.rec$alpha[,i],
            sqrt(m.rec$valpha[,i]),xlab=colnames(xmat)[i])
            if (i> 1) abline(h = 0, col = "gray60",lty =2)}
      title("Number of Exacerbations - ISIS iPFT Subjects with acceptable
      baseline measurements - unexponentiated coefficients",outer=TRUE)

## these exponentiated coefficient plots correspond to those presented in
      par(mfrow=c(2,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      colnames(xmat) <- c("Mean number of exacerbations for males in IS arm
      with mean FEV 0.5, age, height at baseline",
      "Ratio of mean number of exacerbations comparing HS to IS",
      "Ratio of mean number of exacerbations comparing one SD difference in
      baseline FEV 0.5",
      "Ratio of mean number of exacerbations comparing one month difference
      in baseline age",
      "Ratio of mean number of exacerbations comparing one cm difference in
      baseline height",
      "Ratio of mean number of exacerbations comparing females to males" )
      for(i in 1:ncol(xmat)) {ci.plot(m.rec$tis, m.rec$alpha[,i],
            sqrt(m.rec$valpha[,i]),xlab=colnames(xmat)[i],
            fun=exp,dfun=exp)
            if (i> 1) abline(h = 1, col = "gray60",lty =2)}
      title("Number of Exacerbations - ISIS iPFT Subjects with acceptable
      baseline measurements", outer=TRUE)


## construct temporal process response for accumulative days in exacerbation
      do1.acc <- function(x) {
      gap <- x$iv2 - x$iv1 + 1
      if (all(is.na(gap))) yy <- tt <- NULL
      else {
            gap <- na.omit(gap)
            yy <- cumsum(rep(1, sum(gap)))
            tt <- unlist(sapply(1:length(gap), function(i) seq(x$iv1[i],
            x$iv2[i], by=1.0)))
      }
      yy <- c(0, yy, rev(yy)[1])
      tt <- c(0, tt, max(x$futime))
      as.lgtdl(data.frame(time=tt, cov=yy))
      }

      yacc <- lapply(split(isis.pft[,c("id", "iv1", "iv2", "futime")],
isis.pft$id), do1.acc)

## TPR function call for the accumulated days in exacerbation process,
## creates the TPR object m.acc
      tis <- 10:357
      m.acc <- tpr(yacc, rt, xmat[,1:ncol(xmat)], list(),
      xmat[,-(1:ncol(xmat)),drop=FALSE], list(),tis=tis,
      w = rep(1, length(tis)), family = gaussian(),
      evstr = list(link = 1, v = 1))

## plots of the time-varying coefficients from the accumulated days in
## exacerbation TPR fit
      par(mfrow=c(2,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
```

```
colnames(xmat) <- c("Mean days in exacerbation for males in IS arm with
mean FEV 0.5, age, height at baseline",
"Difference in mean days in exacerbation comparing HS to IS",
"Difference in mean days in exacerbation comparing one SD difference in
baseline FEV 0.5",
"Difference in mean days in exacerbation comparing one month difference
in baseline age",
"Difference in mean days in exacerbation comparing one cm difference in
baseline height",
"Difference in mean days in exacerbation comparing females to males")
for(i in 1:ncol(xmat)) {ci.plot(m.acc$tis, m.acc$alpha[,i],
      sqrt(m.acc$valpha[,i]),xlab=colnames(xmat)[i])
      if (i> 1) abline(h = 0, col = "gray60",lty =2)}
title("Accumulated Days in Exacerbation - ISIS iPFT Subjects with
acceptable baseline measurements",outer=TRUE)
```

12. Appendix D – Poisson Process Dataset Construction Code:

12.1. R Code to Construct Data and Apply TPR

```
## This R code constructs data sets that show potential results using the
## extensions of the Cox proportional hazards model and TPR subject to being
## similar to the ISIS data with respect to rate of events and distribution
## of the times to first events

## load required packages
      library(NHPoisson)
      library(survival)
      library(KMsurv)
      library(tpr)

## simulate a 336 day period for first events under the assumption of the
## same (constant) intensity for both treatment arms for the entire 336 day
## duration initiate objects to hold the first event time simulations
      treat.first <- NULL
      cntrl.first <- NULL

## indicate n.subjects per arm, the code below assumes balanced treatment
## arms, 150 per arm will give us roughly the same number of subjects as the
## ISIS trial had
      n.subjects <- 150
## specify lambda for the homogeneous Poisson process event times (2.3 events
## per person-year)
      lambda <- 2.3/365.24

## simulate the event times for each subject in the treatment arm (this code
## is setup so one cound vary itensities by treatment arm if desired) and
## create a data frame with columns for the subject id, event start time,
## simulated event duration event count (total events for the subject), and
## treatment status
      for (i in 1:n.subjects){
            event.times <- simNHP.fun(lambda=c(rep(lambda,336)))[[1]]
            if (length(event.times) == 0) subj.dat <- cbind(i,NA,NA,0)
            else {event.count <- length(event.times)
            event.durs <- round(rweibull(event.count,shape=2,scale=17))
            subj.dat <- cbind(rep(i,event.count),
                  event.times,
                  event.durs,
                  event.count)
            }
            treat.first <- rbind(treat.first,subj.dat)
      }

      treat.first <- as.data.frame(treat.first)
      treat.first <- cbind(treat.first,rep(1,nrow(treat.first)))

##  give the simulated treatment arm data frame column names
      colnames(treat.first) <- c("id","start","duration","count","treatment")


## simulate the event times for each subject in the control arm and create a
```

```
## data frame with columns for the subject id, event start time, simulated
## event duration event count (total events for the subject), and treatment
## status
      for (i in (n.subjects+1):(2*n.subjects)){
            event.times <- simNHP.fun(lambda=c(rep(lambda,336)))[[1]]
            if (length(event.times) == 0) subj.dat <- cbind(i,NA,NA,0)
            else {event.count <- length(event.times)
            event.durs <- round(rweibull(event.count,shape=2,scale=17))
            subj.dat <- cbind(rep(i,event.count),
                  event.times,
                  event.durs,
                  event.count)
            }
            cntrl.first <- rbind(cntrl.first,subj.dat)
      }
      cntrl.first <- as.data.frame(cntrl.first)
      cntrl.first <- cbind(cntrl.first,rep(0,nrow(cntrl.first)))

##  give the simulated control arm data frame column names
      colnames(cntrl.first) <- c("id","start","duration","count","treatment")


## create the first.events data frame that contains *only* the first events
## from both arms generated under the constant intensity Poisson process
      first.events <- rbind(treat.first[!duplicated(treat.first$id),],
            cntrl.first[!duplicated(cntrl.first$id),])


## now simulate a  336 day period for events after the first using the
## non-homogeneous Poisson process, the different intensity functions for
## each treatment arm are specified below specify the two time-dependent
## itensity functions for the non-homogeneos poisson process, these intensity
## functions were obtained via the iterative process described in Section 6.2
      decr.intensity <- function(x) (x-336)^2*(2.1/12644352)*(1.04)
      incr.intensity <- function(x) (2)*(2.1/12644352)*(x^2)


## initiate objects to hold the simulations for the event times after the
## first
      treatment <- NULL
      control <- NULL


## simulate the event times for each subject in the treatment arm and create
## a data frame with columns for the subject id, event start time, simulated
## event duration event count (total events for the subject), and treatment
## status
      for (i in 1:n.subjects){
            event.times <- simNHP.fun(lambda=incr.intensity(1:336))[[1]]
            if (length(event.times) == 0) subj.dat <- cbind(i,NA,NA,0)
            else {event.count <- length(event.times)
            event.durs <- round(rweibull(event.count,shape=2,scale=17))
            subj.dat <- cbind(rep(i,event.count),
                  event.times,
                  event.durs,
                  event.count)
```

```
            }
            treatment <- rbind(treatment,subj.dat)
      }
      treatment <- as.data.frame(treatment)
      treatment <- cbind(treatment,rep(1,nrow(treatment)))
      colnames(treatment) <- c("id","start","duration","count","treatment")

## simulate the event times for each subject in the control arm and create a
## data frame with columns for the subject id, event start time, simulated
## event duration event count (total events for the subject), and treatment
## status
      for (i in (n.subjects+1):(2*n.subjects)){
            event.times <- simNHP.fun(lambda=decr.intensity(1:336))[[1]]
            if (length(event.times) == 0) subj.dat <- cbind(i,NA,NA,0)
            else {event.count <- length(event.times)
            event.durs <- round(rweibull(event.count,shape=2,scale=17))
            subj.dat <- cbind(rep(i,event.count),
                  event.times,
                  event.durs,
                  event.count)
            }
            control <- rbind(control,subj.dat)
      }
      control <- as.data.frame(control)
      control <- cbind(control,rep(0,nrow(control)))
      colnames(control) <- c("id","start","duration","count","treatment")


## append the "treatment" and "control" datasets for events after the first
## into the "dat" matrix
      dat <- NULL
      dat <- rbind(treatment,control)


## increase start times of all events from the varying intensity simulations
## by the start time of the first event from the constant intensity
## simulation
      for (i in first.events$id){
      dat[dat$id == first.events$id[i],]$start <- dat[dat$id ==
      first.events$id[i],]$start + first.events$start[first.events$id == i]
      }

## append the "first.events" data frame to the "dat" frame and sort by id and
## start day so events are sorted by subject id and then their start time in
## ascending order
      dat <- rbind(dat,first.events)
      dat <- dat[order(dat$id,dat$start,decreasing=FALSE),]

## removing any records that now exceed the 336 day end of "follow up" and
## excess NA records these records occured if a subject did not have a first
## event since all subsequent event times were adjusted forward by the first
## event start time (which is NA when none occurred)
      dat <- dat[dat$start <= 336 | is.na(dat$start),]
      dat <- dat[!(is.na(dat$start) & duplicated(dat$id)),]

## append two columns with the "stop" day of events and an indicator of
```

```
## within subject exacerbation number, and give the dat data frame column
## names
      dat <- cbind(dat,dat$start+dat$duration,numeric(nrow(dat)))
      colnames(dat) <- c("id","start","duration","count",
      "treatment","stop","enum")


## combining events that overlap as described in Section 6.2, the overlap
## indicator notes records to be dropped
      overlap <- numeric(nrow(dat))
      for(i in 1:(nrow(dat)-1)){
      if ((dat$stop[i] >= dat$start[i+1]) & (dat$id[i] == dat$id[i+1])){
            dat$stop[i] <- dat$stop[i+1]
            dat$duration[i] <- dat$stop[i]-dat$start[i]
            overlap[i+1] <- 1
      }}
      dat <- dat[overlap==0,]


## populate the enum variable (within subject event count)
      dat$enum[1] <- 1
      for (i in 2:nrow(dat)){
      if(dat$id[i] != dat$id[i-1]) dat$enum[i] <- 1
      else dat$enum[i] <- dat$enum[i-1] + 1
      }

## add a status variable to indicate records associated with events (vs.
## censored)
      dat <- cbind(dat,as.numeric(!is.na(dat$start)))
      colnames(dat) <- c("id","start","duration","count",
      "treatment","stop","enum","status")

## updating the count variable to reflect the combination of overlapping
## events
      for (i in unique(dat$id)){
      dat$count[dat$id == i] <- nrow(dat[dat$id == i &dat$status == 1,])
      }

## add a follow up time column to "dat", follow up is specified to be 336
## days for all but the subjects who began an event before 336 days that
## lasted beyond the 336 day time point (see next comments/code for
## addressing those cases)
      futime <- rep(336,times=nrow(dat))
      dat <- cbind(dat,futime)

## extend the follow up time for the specific records for events that last
## beyond 336 days once their duration is accounted for the following for
## loop extends the follow up time for ALL records for those individuals
      dat$futime[!is.na(dat$stop) & dat$stop > dat$futime] <-
            dat$stop[!is.na(dat$stop) & dat$stop > dat$futime]
      for (i in (dat$id[dat$futime > 336])){
            dat$futime[dat$id == i] <- max(dat$futime[dat$id == i])
      }


## extract first events to fit simple cox model
```

```
      first.events.dat <- NULL
      first.events.dat <- dat[dat$enum == 1,]

## rename start as "time" and set this to 336 for subjects with no event(s)
## as this is the "follow up" for subjects with no event(s)
      colnames(first.events.dat)[colnames(first.events.dat) == "start"] <-
            "time"
      first.events.dat$time[is.na(first.events.dat$time)] <- 336


## plot the Kaplan-Meier survival curves associated with the first events
## only, print on this plot the hazard ratio and p-value from a Cox
## proportional hazards model fit for this first event only data, as well as
## the rates of events per person-years in the two arms and a p-value from a
## Poisson regression comparing those rates
      my.surv <- Surv(time=first.events.dat$time,
            event=first.events.dat$status)
      my.fit <- survfit(my.surv~first.events.dat$treatment)
      summary(pfit <- glm(count ~ treatment, family = "poisson",
            data = first.events.dat,offset = rep(log(336/365.24),times=300)))
      par(mfrow=c(1,1))
## uncomment the following "par" call to put this plot as well as the TPR
## fits for this data on the same page, also comment out "par" calls
## identified at the end of the TPR code

      #par(mfcol=c(2,3), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,0,0))
      plot(my.fit)
      text(0,0, paste("control rate:",round(exp(pfit$coef[1]),2),
            "  treat rate:",round(exp(pfit$coef[1]+pfit$coef[2]),2),
            "  p-val:",round(summary(pfit)$coef[2,4],2)),pos=4)
      text(336,1, paste("Cox HR:", round(exp(summary(
            coxph(my.surv~first.events.dat$treatment))$coef[1]),2),
            "p-val:", round(summary(coxph(my.surv~first.events.dat$treatment)
             )$coef[5],2)),pos=2)

      plot.new()


## TPR Code - refer to appendix B or C if more detail is needed

## rename a few columns so the variable names match those in this TPR code
      colnames(dat)[colnames(dat) == "treatment"] <- "rx"
      colnames(dat)[colnames(dat) == "start"] <- "iv1"
      colnames(dat)[colnames(dat) == "stop"] <- "iv2"



## construct temporal process response for total number of exacerbations
      sim.data <- unique(dat[,c("id", "futime", "rx")])

      rec <- lapply(split(dat[,c("id", "iv1", "futime")], dat$id),
      function(x) {
            v <- x$iv1
            maxfu <- max(x$futime)
            if (is.na(v[1])) c(0, maxfu)
            else c(0, v[!is.na(v)], maxfu)
```

```
})

yrec <- lapply(rec,
function(x) {
      sim.data <- data.frame(time=x, cov=1:length(x)-1)
      len <- length(x)
      sim.data$cov[len] <- sim.data$cov[len - 1]
      as.lgtdl(sim.data)
})

tu <- max(sim.data$futime)
rt <- lapply(1:nrow(sim.data),
function(i) {
      x <- sim.data[i, "futime"]
      time <- c(0, x, tu)
      cov <- c(1, 0, 0)
      as.lgtdl(data.frame(time=time, cov=cov))
      })



xmat <- model.matrix(~ rx, data=sim.data)


tis <- 10:336
m.rec <- tpr(y=yrec,delta=rt,x=xmat[,1:2],xtv=list(),z=xmat[,-(
1:2),drop=FALSE],tis=tis, w = rep(1, length(tis)), family = poisson(),
evstr = list(link = 5, v = 3))


## construct temporal process response for accumulative days exacerbation
do1.acc <- function(x) {
gap <- x$iv2 - x$iv1 + 1
if (all(is.na(gap))) yy <- tt <- NULL
else {
      gap <- na.omit(gap)
      yy <- cumsum(rep(1, sum(gap)))
      tt <- unlist(sapply(1:length(gap), function(i) seq(x$iv1[i],
      x$iv2[i], by=1.0)))
      }
yy <- c(0, yy, rev(yy)[1])
tt <- c(0, tt, max(x$futime))
as.lgtdl(data.frame(time=tt, cov=yy))
}

yacc <- lapply(split(dat[,c("id", "iv1", "iv2", "futime")], dat$id),
do1.acc)

xmat <- model.matrix(~ rx, data=sim.data)

tis = 10:330
m.acc <- tpr(yacc, rt, xmat[,1:2], list(), xmat[,-(1:2),drop=FALSE],
list(),tis=tis, w = rep(1, length(tis)), family = gaussian(),
evstr = list(link = 1, v = 1))
```

```
## create plots of all four coefficients over time (both intercept and
## treatment) from both models (mean events and mean days in episodes)
## comment out the "par" call lines if plotting all results on the same page
## (see code that plots K-M curves)

      colnames(xmat) <- c("Mean number of events for control arm",
      "Ratio of mean number of events comparing treatment to control")
      par(mfrow=c(1,2), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      for(i in 1:2) {ci.plot(m.rec$tis, m.rec$alpha[,i],
            sqrt(m.rec$valpha[,i]),fun=poisson()$linkinv,
            dfun = poisson()$mu.eta,  xlab=colnames(xmat)[i])
            if (i> 1) abline(h = 1, col = "gray60",lty =2)
      }
      title("Number of Events - Simulated Data", outer=TRUE)

      colnames(xmat) <- c("Mean days in episodes for control arm",
      "Diff. in mean days in episodes comparing treatment to control")
      par(mfrow=c(1,2), mgp=c(2,1,0), mar=c(4,2,1,0), oma=c(0,2,2,0))
      for(i in 1:2) {ci.plot(m.acc$tis, m.acc$alpha[,i],
            sqrt(m.acc$valpha[,i]),xlab=colnames(xmat)[i])
      if (i> 1) abline(h = 0, col = "gray60",lty =2)
      }
      title("Accumulated Days in Episodes - Simulated Data", outer=TRUE)


## export data frame to Stata .dta format for application of extensions of
## the Cox model see following Stata code
      library(foreign)
      write.dta(dat, "sim.dta")
```

12.2. **Stata Code to Apply Extensions of the Cox Model to Constructed Data**

```
** Stata code for the application of AG/Conditional and Marginal extensions
** of the Cox model
      use "sim.dta"

      rename iv1 start
      rename iv2 stop
      drop enum
      rename rx trt

      save "temp.dat",replace

* adding 'final' observations for subjects who had at least one event
      drop if status == 0

      sort id start
      bysort id: keep if _n == _N
      replace start = .
      replace stop = .
      replace status = 0

      append using "temp.dat"
      sort id start


* gen time1, time2 and enum to represent the time entering the risk set, time
* leaving and within id event counter
      bysort id: gen enum = _n
      gen time1 = 0
      gen time2 = start
* if time2 is missing, the subject never had an event so their time exiting
* the risk set is their follow up
      replace time2 = futime if start == . & stop == .
* update 'time1' to be the end of a subject's previous event (i.e. when they
* reenter the risk set)
      bysort id: replace time1 = stop[_n-1] if _n != 1


* dropping ivstart and ivstop variables
      drop start stop

* drop added 'final' records that don't add any time at risk
      drop if time1 >= time2 & time1 >= futime & status == 0

      save "AG-conditional format.dta", replace


*** Checking Exacerbation counts - note very few subjects with > 3 events

      bysort id: gen num_exacerbations = sum(status)
      bysort id: keep if _n == _N
      sort id enum
      tab trt num_exacerbations
```

```
*** Poisson regression
      gen log_obstime = log(336)
      poisson num_exacerbations trt, irr offset(log_obstime)


* Reopen the prepared dataset
      clear
      use "AG-conditional format.dta"

*** Below are the "First event", "Anderson-Gill", and two "Conditional"
*** analyses

*** Time to first analysis
      stset time2 if enum == 1, failure(status) time0(time1)
      *stcox trt, nolog nohr robust
      stcox trt, nolog robust


* KM plot by treatment group
      sts graph, by(trt) legend(label(1 "Control") label(2 "Treatment")) ///
      tmax(340) riskt(0 84 168 252 336, order(1 "Treatment" 2 "Control")) ///
      title("Kaplan-Meier survival estimates - constructed data", ///
      size(med)) plot2opts(lpattern(dash))///
       xtitle("Days since randomization") xlabel(0 84 168 252 336)


*** Anderson-Gill
* time 2 is event time, time1 enrollment time
      stset time2, failure(status) time0(time1)

      *stcox trt, cluster(id) nolog nohr
      stcox trt, cluster(id) nolog


*** Conditional
*stcox trt, cluster(id) strata(enum) nolog nohr
      stcox trt, cluster(id) strata(enum) nolog

** generating collpased enum variables, one for 3 levels as used in
** Therneau's DNase analysis and one with 5
      recode enum (1=1) (2=2) (3=3) (4=4) (5/10=5), generate(enum_5)
      recode enum (1=1) (2=2) (3/10=3), generate(enum_3)


* Conditional/3
      stcox trt, cluster(id) strata(enum_3) nolog


* Conditional/5
      stcox trt, cluster(id) strata(enum_5) nolog


* Conditional/3 with treatment interaction based on strata
      stcox  c.trt#i.enum_3, cluster(id) strata(enum_3) nolog
```

```
* Conditional/5 with treatment interaction based on strata
      stcox  c.trt#i.enum_5, cluster(id) strata(enum_5) nolog



* Data manipulation to create the appropriate risk sets for application of
* the WLW Marginal Model keep all of the records for "first" at risk strata,
* this case is fairly straightforward
      clear
      cd "C:\Users\Arthur Baines\Dropbox\Thesis Work\Simulation Results"
      use "AG-conditional format.dta"
      keep if enum == 1
      save "temp1.dta", replace

* setting up records for the "second" at risk strata, we want to keep all
* records for the first or second strata and any first strata record ending
* in an event needs it's status reset since it is not an event for strata two
* lastly, the records we are keeping from strata 1 must be relabeled since we
* want them to be attributed to *strata 2
      clear
      use "AG-conditional format.dta"
      keep if enum < 3
      replace status = 0 if enum < 2
      replace enum = 2
      save "temp2.dta", replace

* setting up records for the "third" at risk strata, we want to keep all
* records for the first through third strata and any first/second strata
* record ending in an event needs it's status reset since it is not an event
* for strata 3, lastly, the records we are keeping from strata 1/2 must be
* relabeled since we want them to be attributed to strata 3
      clear
      use "AG-conditional format.dta"
      keep if enum < 4
      replace status = 0 if enum < 3
      replace enum = 3
      save "temp3.dta", replace

* see above but for strata 4
      clear
      use "AG-conditional format.dta"
      keep if enum < 5
      replace status = 0 if enum < 4
      replace enum = 4
      save "temp4.dta", replace

* see above but for strata 5
      clear
      use "AG-conditional format.dta"
      replace status = 0 if enum < 5
      replace enum = 5
      save "temp5.dta", replace

* see above but for strata 6
      clear
```

```
      use "AG-conditional format.dta"
      replace status = 0 if enum < 6
      replace enum = 6
      save "temp6.dta", replace

* see above but for strata 7
      clear
      use "AG-conditional format.dta"
      replace status = 0 if enum < 7
      replace enum = 7
      save "temp7.dta", replace

* see above but for strata 8
      clear
      use "AG-conditional format.dta"
      replace status = 0 if enum < 8
      replace enum = 8
      save "temp8.dta", replace

* see above but for strata 9
      clear
      use "AG-conditional format.dta"
      replace status = 0 if enum < 9
      replace enum = 9
      save "temp9.dta", replace

* see above but for strata 10
      clear
      use "AG-conditional format.dta"
      replace status = 0 if enum < 10
      replace enum = 10
      save "temp10.dta", replace


* only append the temp files up to the maximum number of exacerbations had
* (if the most had was 7, append temp1.dta through temp7.dta)
      clear
      use "temp1.dta"
      append using "temp2.dta"
      append using "temp3.dta"
      append using "temp4.dta"
      append using "temp6.dta"
      append using "temp7.dta"
      append using "temp8.dta"
      append using "temp9.dta"
      append using "temp10.dta"
      save "WLW-marginal format.dta", replace

      clear
      use "WLW-marginal format.dta"


* WLW
      stset time2, failure(status) time0(time1)
      stcox trt, cluster(id) strata(enum) nolog
```

```
** generating a collpased enum variable
      recode enum (1=1) (2=2) (3=3) (4=4) (5/10=5), generate(enum_5)
      recode enum (1=1) (2=2) (3/10=3), generate(enum_3)

* WLW/3
      stcox trt, cluster(id) strata(enum_3) nolog

* WLW/5
      stcox trt, cluster(id) strata(enum_5) nolog

* WLW/3 with treatment interaction based on strata
      stcox  c.trt#i.enum_3, cluster(id) strata(enum_3) nolog

* WLW/5 with treatment interaction based on strata
      stcox  c.trt#i.enum_5, cluster(id) strata(enum_5) nolog
```

13. **Appendix E – Kelly and Lim Simulation Recreation Code:**

```
## This R code reproduces a subset of the simulations carried out by Kelly &
## Lim (2000) in their paper Survival Analysis for Recurrent Event Data: An
## Application to Childhood Infectious Diseases

## load required packages
      library(survival)
      library(matrixStats)


## initiate three objects for storing estimates from three models we are
interested in
## here: the AG, Conditional common treatment effect, and Conditional with
## treatment/event interaction (event specific treatment estimate)
      ag.est <- NULL
      pwp.est <-NULL
      pwp.eventspec.est <- NULL

## simulate data under K&L scenario (i) with constant treatment effect and
## zero within-subject correlation (sigma.2 = 0). Rerun this entire code
## using the commented out set of betas to reproduce K&L scenario (ii)
      beta.0 <- 3
      beta.1 <- 1
      beta.2 <- 1
      beta.3 <- 1
      beta.4 <- 1

#     beta.0 <- 3
#     beta.1 <- 1
#     beta.2 <- 0
#     beta.3 <- 0
#     beta.4 <- 0

## the within-subject correlation version of K&L's simulations can be
## reproduced by changing the sigma value below
      sigma.2 <- 0
      v <- rnorm(2*n,mean=0,sd=sqrt(sigma.2))

## create an "expected beta" vector indicating the true betas under the
## common or event specific models
      exp.beta <- -1*c(beta.1,beta.2,beta.3,beta.4)
      exp.beta.common <- (sum(exp.beta)/4)


## set desired number of simulations, K&L use 100 simluations of the 500
## subject datasets
      n.sim <- 100

## set the desired number of subjects per arm (this code assumed balanced
## arms)
      n <- 250

## loop over the code that simulations each of the n.sim datasets
      for (i in 1:n.sim){
```

```
## simulate the 4 independent event times for the treatment arm subjects,
## these event times indicate the time since the previous event (or time 0
## for the first event)
      trt.1 <- rexp(n)*exp(beta.0 + beta.1 + v[1:n])
      trt.2 <- rexp(n)*exp(beta.0 + beta.2 + v[1:n])
      trt.3 <- rexp(n)*exp(beta.0 + beta.3 + v[1:n])
      trt.4 <- rexp(n)*exp(beta.0 + beta.4 + v[1:n])

## create a matrix for treatment subjects with columns of: id, treatment
## status, time0 and time1 this matrix has 4 rows for each subject id,
## corresponding to their 1st through 4th event times the time0 variable
##indicates the time of re-entry into the risk set in days from time zero
## the time1 variable indicates the time of the event in days from time zero
      trt <- cbind(rep(1:250,times=4),
            rep(1,times=(4*n)),
            rbind(cbind(rep(0,times=n),trt.1),
            cbind(trt.1,trt.1+trt.2),
            cbind(trt.1+trt.2,trt.1+trt.2+trt.3),
            cbind(trt.1+trt.2+trt.3,trt.1+trt.2+trt.3+trt.4)))

## reorder the trt matrix by subject id
      trt <- trt[order(trt[,1]),]

## simulate the 4 independent event times for the control arm subjects, these
## event times indicate the time since the previous event (or time 0 for the
## first event)
      ctrl.1 <- rexp(n)*exp(beta.0 + v[(n+1):(2*n)])
      ctrl.2 <- rexp(n)*exp(beta.0 + v[(n+1):(2*n)])
      ctrl.3 <- rexp(n)*exp(beta.0 + v[(n+1):(2*n)])
      ctrl.4 <- rexp(n)*exp(beta.0 + v[(n+1):(2*n)])

## create a matrix for control subjects with columns of: id, treatment
## status, time0 and time1 this matrix has 4 rows for each subject id,
## corresponding to their 1st through 4th event times the time0 variable
## indicates the time of re-entry into the risk set in days from time zero
## the time1 variable indicates the time of the event in days from time zero
      ctrl <- cbind(rep((n+1):(2*n),times=4),
            rep(0,times=(4*n)),
            rbind(cbind(rep(0,times=n),ctrl.1),
            cbind(ctrl.1,ctrl.1+ctrl.2),
            cbind(ctrl.1+ctrl.2,ctrl.1+ctrl.2+ctrl.3),
            cbind(ctrl.1+ctrl.2+ctrl.3,ctrl.1+ctrl.2+ctrl.3+ctrl.4)))

## reorder the ctrl matrix by subject id
      ctrl <- ctrl[order(ctrl[,1]),]

## create one data frame "sim.dat" with both treatment and control arm event
times
      sim.dat <- as.data.frame(rbind(trt,ctrl))

## add a column indicating event status (1 for event, 0 for censored), and
two variables to
## be populated later: enum (event num) and count (total events)
      sim.dat <- cbind(sim.dat,rep(1,times=4*n),
            rep(NA,times=4*n),rep(NA,times=4*n))
```

```
## give the sim.dat data frame names
      names(sim.dat) <- c("id","trt","time0","time1","status","enum","count")

## drop records associated with at risk intervals that begin after day 120
      sim.dat <- sim.dat[!(sim.dat$time0 > 120),]

## set event status to zero (no event) for records with event times after day
## 120
      sim.dat$status[sim.dat$time1 > 120] <- 0
      sim.dat$time1[sim.dat$time1 > 120] <- 120

## populate the count variable
      for (i in unique(sim.dat$id)){
            sim.dat$count[sim.dat$id == i] <- nrow(sim.dat[sim.dat$id == i
            &sim.dat$status == 1,])
      }


## populate the enum variable
      sim.dat$enum[1] <- 1
      for (i in 2:nrow(sim.dat)){
            if(sim.dat$id[i] != sim.dat$id[i-1]) sim.dat$enum[i] <-  1
            else sim.dat$enum[i] <- sim.dat$enum[i-1] + 1
      }

## reset rownames for sim.dat
      rownames(sim.dat) <- NULL

## create treatment/event num interactions for fitting interaction models
      sim.dat <- cbind(sim.dat,sim.dat$trt*(sim.dat$enum==1))
      sim.dat <- cbind(sim.dat,sim.dat$trt*(sim.dat$enum==2))
      sim.dat <- cbind(sim.dat,sim.dat$trt*(sim.dat$enum==3))
      sim.dat <- cbind(sim.dat,sim.dat$trt*(sim.dat$enum==4))

      colnames(sim.dat)[8:11] <- c("trt1","trt2","trt3","trt4")


## Anderson-Gill model fit
      fita <- coxph(Surv(time0, time1, status) ~ trt + cluster(id),
             data=sim.dat)

## Conditional model fit (common treatment effect)
      fitc <- coxph(Surv(time0, time1, status) ~ trt + cluster(id) +
            strata(enum), data=sim.dat, robust=TRUE)

## Conditional model fit (allowing for event specific treatment effect)
      fitc3 <- coxph(Surv(time0, time1, status) ~ trt1 + trt2 + trt3 + trt4 +
            cluster(id) + strata(enum), data=sim.dat, robust=TRUE)

## store estimates from these fits in the model specific estimate matrices
      ag.est <- rbind(ag.est,c(log(summary(fita)$conf.int[,-2]),
             sqrt(fita$naive.var),sqrt(fita$var)))
      pwp.est <- rbind(pwp.est,c(log(summary(fitc)$conf.int[,-2]),
            sqrt(fitc$naive.var),sqrt(fitc$var)))
```

```
      pwp.eventspec.est <- rbind(pwp.eventspec.est,
            c(as.vector(log(summary(fitc3)$conf.int[,-2])),
            sqrt(diag(fitc3$naive.var)),
            sqrt(diag(fitc3$var))))
      }

## convert the model specific estimate matrices to data frames
      ag.est <- as.data.frame(ag.est)
      pwp.est <- as.data.frame(pwp.est)
      pwp.eventspec.est <- as.data.frame(pwp.eventspec.est)

## name the model specific matrix columns for easier reference
      colnames(ag.est) <- c("beta","ci.lower","ci.upper","se.naive",
            "se.robust")
      colnames(pwp.est) <- c("beta","ci.lower","ci.upper","se.naive",
            "se.robust")
      colnames(pwp.eventspec.est) <- c("beta1","beta2","beta3","beta4",
            "ci.lower1","ci.lower2","ci.lower3","ci.lower4",
            "ci.upper1","ci.upper2","ci.upper3","ci.upper4",
            "se.naive1","se.naive2","se.naive3","se.naive4",
            "se.robust1","se.robust2","se.robust3","se.robust4")


## create and (name) data frame to hold the final simulations results
      results <- as.data.frame(matrix(NA,nrow=6,ncol=7))
      colnames(results) <- c("betahat.mean","betahat.sd","betahat.bias",
            "se.naive.mean","se.robust.mean","cover.naive","cover.robust")
      rownames(results) <- c("ag.common","pwp.common","pwp.eventspec.1",
            "pwp.eventspec.2","pwp.eventspec.3","pwp.eventspec.4")

## populate the AG simulation results for the common treatment effect model
      results[1,] <- c(round(c(colMeans(ag.est)[1],colSds(ag.est)[1],
      colMeans(ag.est)[1]-(exp.beta.common),
      colMeans(ag.est)[(4:5)]),3),
      nrow(ag.est[(ag.est$beta - ag.est$se.naive*1.96) <
            exp.beta.common & (ag.est$beta + ag.est$se.naive*1.96) >
            exp.beta.common,])/n.sim,
      nrow(ag.est[(ag.est$beta - ag.est$se.robust*1.96) <
            exp.beta.common & (ag.est$beta + ag.est$se.robust*1.96) >
            exp.beta.common,])/n.sim)

## populate the Conditional simulation results for the common treatment
## effect model
      results[2,] <- c(round(c(colMeans(pwp.est)[1],colSds(pwp.est)[1],
      colMeans(pwp.est)[1]-(exp.beta.common),
      colMeans(pwp.est)[(4:5)]),3),
      nrow(pwp.est[(pwp.est$beta - pwp.est$se.naive*1.96) <
            exp.beta.common & (pwp.est$beta + pwp.est$se.naive*1.96) >
            exp.beta.common,])/n.sim,
      nrow(pwp.est[(pwp.est$beta - pwp.est$se.robust*1.96) <
            exp.beta.common & (pwp.est$beta + pwp.est$se.robust*1.96) >
            exp.beta.common,])/n.sim)

## populate the Conditional event specific model result fields
      results$betahat.mean[3:6] <- round(colMeans(pwp.eventspec.est[1:4]),3)
      results$betahat.sd[3:6] <- round(colSds(pwp.eventspec.est[1:4]),3)
```

```
      results$betahat.bias[3:6] <- round(colMeans(pwp.eventspec.est[1:4])-
            (exp.beta),3)
      results$se.naive.mean[3:6] <- round(
            colMeans(pwp.eventspec.est)[13:16],3)
      results$se.robust.mean[3:6] <- round(
            colMeans(pwp.eventspec.est)[17:20],3)
      results$cover.naive[3:6] <- c(sum((pwp.eventspec.est[1] -
            pwp.eventspec.est[13]*1.96) < exp.beta[1] &
            (pwp.eventspec.est[1] + pwp.eventspec.est[13]*1.96) >
            exp.beta[1])/n.sim,
            sum((pwp.eventspec.est[2] - pwp.eventspec.est[14]*1.96) <
            exp.beta[2] & (pwp.eventspec.est[2] + pwp.eventspec.est[14]*1.96)
             > exp.beta[2])/n.sim,
            sum((pwp.eventspec.est[3] - pwp.eventspec.est[15]*1.96) <
            exp.beta[3] & (pwp.eventspec.est[3] + pwp.eventspec.est[15]*1.96)
            > exp.beta[3])/n.sim,
            sum((pwp.eventspec.est[4] - pwp.eventspec.est[16]*1.96) <
            exp.beta[4] & (pwp.eventspec.est[4] + pwp.eventspec.est[16]*1.96)
            > exp.beta[4])/n.sim)
      results$cover.robust[3:6] <- c(sum((pwp.eventspec.est[1] -
            pwp.eventspec.est[17]*1.96) < exp.beta[1] &
            (pwp.eventspec.est[1] + pwp.eventspec.est[17]*1.96) >
            exp.beta[1])/n.sim,
            sum((pwp.eventspec.est[2] - pwp.eventspec.est[18]*1.96) <
            exp.beta[2] & (pwp.eventspec.est[2] + pwp.eventspec.est[18]*1.96)
            > exp.beta[2])/n.sim,
            sum((pwp.eventspec.est[3] - pwp.eventspec.est[19]*1.96) <
            exp.beta[3] & (pwp.eventspec.est[3] + pwp.eventspec.est[19]*1.96)
            > exp.beta[3])/n.sim,
            sum((pwp.eventspec.est[4] - pwp.eventspec.est[20]*1.96) <
            exp.beta[4] & (pwp.eventspec.est[4] + pwp.eventspec.est[20]*1.96)
            > exp.beta[4])/n.sim)
      results <- cbind(c(exp.beta.common,exp.beta.common,exp.beta),results)
      colnames(results)[1] <- "beta.true"

## print the results data frame
      results
```