COVID – Evaluating Treatment Efficacy in Hospitalized Covid-19 Patients

Yu Gu, Jianqiao Wang, Donglin Zeng, and Dan-Yu Lin

November 6, 2022

Introduction

COVID implements the two methods of Lin et al. (2022) to assess the totality of evidence for the treatment effects in Covid-19 clinical trials. Both methods draw a single probabilistic conclusion about the treatment effects on the entire clinical course of a patient through full use of daily clinical status. The first method specifies proportional hazards models for the times to all possible levels of improvement or deterioration in clinical status, while the second method specifies proportional odds models for the clinical status of each day.

COVID inputs a data set with the following information:

- **Subject ID**: The variable used to identify subjects.
- **Treatment**: The variable indicating the treatment arm.
- **Initial Status**: The clinical status at randomization.
- **Examination Time**: The times when the subject is examined for clinical status.
- **Clinical Status**: The clinical status of the subject at each examination.
- **Covariates**: Baseline covariates (e.g., baseline disease severity).

Of note, an arbitrary number of baseline covariates can be included, and the examination times are measured from the start of the trial and are specified in units of whole days.

The primary analysis tools of the package are *ph()* and *po()*. *ph()* returns the estimated hazard ratio for each level of improvement or deterioration from the clinical status at enrollment, the estimate of a common hazard ratio for any level of improvement, the estimate of a common hazard ratio for any level of deterioration, and the estimate of a common hazard ratio summarizing the overall treatment benefit, together with the 95% confidence interval and the two-sided p-value for testing no treatment effect on each of the endpoints. *po()* returns the estimated common odds ratio of the lower severity over the entire clinical trial, estimated piecewise log-linear odds ratio of lower severity, and estimated daily odds ratios of lower severity. The 95% confidence intervals are also provided.

In addition, the package includes a convenience function: *outcome()*, which is used to wrap examination time and clinical status together and is part of the model statement of *ph()* and *po()*.

Finally, a toy dataset *acttData* is provided to illustrate the use of the software.

Functions

outcome()

This convenience function is used as the left-hand side of a formula object for the sole purpose of simplifying the specification of examination time and clinical status. This function is not intended to be used as a stand-alone feature. For completeness, the function ensures that the input data obey basic constraints and returns the data in a predictable format for use in internal functions.

The usage is

```
outcome(examination.time, clinical.status)
```

where `examination.time` is the time when a subject is examined for clinical status (NA if missing); and `clinical.status` is the clinical status at each examination (NA if missing).

ph()

This function is one of the primary tools of **COVID**. The matrix returned contains the estimated hazard ratio in terms of treatment versus placebo for each level of improvement or deterioration from the clinical status at enrollment, the estimate of a common hazard ratio for any level of improvement, the estimate of a common hazard ratio for any level of deterioration, and the estimate of a common hazard ratio summarizing the overall treatment benefit, together with the 95% confidence interval and the two-sided p-value for testing no treatment effect on each of the endpoints.

The function call takes the following form:

```
ph(formula, data, subject,
   treatment, init.status, nmin = 5)
```

where

- `formula` is a model statement. See below for further details.
- `data` is a data.frame object containing all required data as previously described.
- `subject` is a character string indicating the variable in data which identifies multiple rows from the same subject.
- `treatment` is a character string indicating the variable in data which corresponds to the treatment arm.
- `init.status` is a character string corresponding to the variable in data which identifies the initial status.
- `nmin` is a positive integer, which is the minimum number of cases for an event to be taken into account. Any event with fewer cases than this number will not be modelled.

The model statement is a formula object. The left side is an object returned by the *outcome()* function and specifies examination time and clinical status. The right side contains treatment arm and all baseline covariates. Categorical baseline covariates can be specified, and all other categories are compared to the first category. A model without baseline covariates is allowed.

The `formula` input takes the following general structure

```
outcome(examination_time, clinical_status) ~ treatment_arm + covariates
```

where `examination_time`, `clinical_status`, `treatment_arm`, and `covariates` are place holders indicating the data that are to be provided; they should be replaced by the appropriate variable names in the header of the input data.

po()

This function is one of the primary tools of **COVID**. The value object returned is a list that contains the estimated common odds ratio of lower severity over a time window, the estimated piecewise log-linear odds ratio of lower severity over a time window, and the daily odds ratio estimates of lower severity.

The usage is

```
po = function(formula, data, subject, treatment,
              imputation = F,
              common.odds.ratio = T,
              piecewise.linear = T,
              intercept = T,
              knots = NULL,
              control.ngd = list(learning.rate = 0.1,
                                 max.iter = 1000,
                                 eps = 1e-6,
                                 messages = F),
              start.time = NULL, end.time = NULL,
              imputed.score = 7)
```

where

- `formula` is a model statement. See *ph* for further details.
- `data` is a data.frame object containing all required data as previously described. The clinical status must contain at least two categories at each examination time.
- `subject` is a character string indicating the variable in data which identifies multiple rows from the same subject.
- `treatment` is a character string indicating the variable in data which corresponds to the treatment arm.
- `imputation` is a logical object that determines whether the input data needs to be imputed by carrying the last observation forward.
- `common.odds.ratio` is a logical object that determines whether the common odds ratio of lower severity over time is estimated.
- `piecewise.linear` is a logical object that determines whether a piecewise log-linear odds ratio over lower severity over time and daily odds ratios of lower severity are estimated. See below for further details.
- `intercept` is a logical object that determines whether intercept is included for the piecewise linear function while assuming a piecewise log-linear odds ratio over time. See below for further details.
- `knots` is a numeric vector that contains the potential change points in days of the piecewise log-linear odds ratio. See below for further details.
- `control.ngd` is a list object that contains hyperparameters for natural gradient ascent algorithm. See below for further details.
- `start.time` is the first examination day included in the analysis.
- `end.time` is the last examination day included in the analysis.
- `imputed.score` is the clinical status used for imputation when there are no measurements for a subject.

The piecewise linear function for the log odds raio is determined by the `intercept` and `knots`. If the intercept is excluded from the piecewise linear function, the model assumes no treatment effect on odds ratio

of lower severity at examination time zero. The `knots` are the change points in the piecewise linear function. For example, if we assume the treatment effect on log odds ratio of lower severity has change points at examination time 0, 8, 15, 21, the input argument should be `knots = c(0, 8, 15, 21)`.

We use natural gradient ascent algorithm to maximize the log likelihood function of the proportional odds model. `control.ngd` contains four hyperparameters in natural gradient ascent algorithm

- `learning.rate` determines the step size for natural gradient ascent algorithm. Must be numeric. The default value is 0.1.
- `max.iter` is the maximum number of iterations for natural gradient ascent algorithm. Must be integer.The default value is 1000.
- `eps` is the convergence threshold for natural gradient ascent algorithm. Must be numeric. The default value if `1e-6`.
- `messages` decides whether to print out the coefficient estimates and log likelihood during optimization. Must be logical. The default value is FALSE.

Examples

To illustrate the call structure and results of `ph` and `po`, we use the dataset provided with the package, `acttData`. The dataset was randomly scrambled from the Adaptive COVID-19 Treatment Trial-1 (ACTT-1), and contains the following observations for each of the 500 patients, followed for up to 29 days:

- **subject_id**: Subject ID.
- **treatment_arm**: A binary indicator of treatment arm (0: placebo; 1: remdesivir).
- **initial_status**: The initial clinical status at randomization.
- **clinical_status**: The clinical status at each examination.
- **examination_time**: The examination time in days.
- **baseline_severity**: The baseline disease severity (Mild-Moderate Disease or Severe Disease).
- **age**: Age in years.
- **sex**: A binary indicator of sex (female or male).

The data can be loaded in the usual way

```
data(acttData)
```

```
head(acttData)
```

```
##   subject_id treatment_arm initial_status clinical_status examination_time
## 1          1             0              6               6                0
## 2          1             0              6               6                1
## 3          1             0              6               5                2
## 4          1             0              6               5                3
## 5          1             0              6               5                4
## 6          1             0              6               5                5
##   baseline_severity sex age
## 1    Severe Disease   F  75
## 2    Severe Disease   F  75
## 3    Severe Disease   F  75
## 4    Severe Disease   F  75
## 5    Severe Disease   F  75
## 6    Severe Disease   F  75
```

Consider the summary statistics

```
summary(acttData)
```

```
##   subject_id        treatment_arm    initial_status  clinical_status
##  Length:8032        Min.   :0.0000   Min.   :4.000   Min.   :1.000
##  Class :character   1st Qu.:0.0000   1st Qu.:5.000   1st Qu.:5.000
##  Mode  :character   Median :0.0000   Median :6.000   Median :6.000
##                     Mean   :0.4955   Mean   :5.892   Mean   :5.426
##                     3rd Qu.:1.0000   3rd Qu.:7.000   3rd Qu.:7.000
##                     Max.   :1.0000   Max.   :7.000   Max.   :8.000
##                                      NA's   :6       NA's   :6
##  examination_time             baseline_severity sex           age
##  Min.   : 0.00    Mild-Moderate Disease: 575    F:2797   Min.   :21.00
##  1st Qu.: 4.00    Severe Disease       :7457    M:5235   1st Qu.:51.00
##  Median : 9.00                                          Median :62.00
##  Mean   :11.04                                          Mean   :60.08
##  3rd Qu.:17.00                                          3rd Qu.:71.00
##  Max.   :29.00                                          Max.   :89.00
##  NA's   :6
```

We can see that the initial clinical status can be 4, 5, 6 or 7. The clinical status ranges from 1 to 8. The data has a total 8032 records for 500 patients.

In the first example, we fit marginal proportional hazards models to evaluate the treatment effects on multiple endpoints defined by different levels of improvement or deterioration in clinical status, as well as the overall benefits of the treatment. The function call takes the following form

```
model <- outcome(examination_time, clinical_status) ~ treatment_arm + baseline_severity
result1 <- ph(formula = model,
              data = acttData,
              subject = "subject_id",
              treatment = "treatment_arm",
              init.status = "initial_status")
```

The function returns a matrix containing the estimated hazard ratio in terms of treatment versus placebo for each endpoint, together with the 95% confidence interval and the two-sided p-value for testing no treatment effect.

```
result1
```

```
##                                    HR Lower .95 Upper .95          P
## Improvement by 1 category     1.0635764 0.8851963  1.277903 0.51049876
## Improvement by 2 categories   1.1763553 0.9709472  1.425219 0.09714095
## Improvement by 3 categories   1.1778374 0.9604798  1.444383 0.11581295
## Improvement by 4 categories   1.2834009 0.9999732  1.647162 0.05002041
## Improvement by 5 categories   1.3572415 0.8978629  2.051655 0.14735454
## Improvement by 6 categories   1.6509813 0.7810405  3.489882 0.18922487
## Any improvement (WLW_imp)     1.2072793 0.9894063  1.473129 0.06358000
## Deterioration by 1 category   0.7469135 0.5394540  1.034156 0.07879950
## Deterioration by 2 categories 0.6760649 0.4106048  1.113147 0.12388291
## Deterioration by 3 categories 0.6671087 0.2645630  1.682148 0.39096960
## Deterioration by 4 categories 2.4193984 0.2334611 25.072649 0.45894050
## Any deterioration (WLW_det)   0.7676759 0.4946771  1.191335 0.23833132
## Overall benefit (WLW_ben)     1.2276618 0.9876743  1.525962 0.06456705
```

In the second example, we estimate the common odds ratio of lower severity over day 1 to day 28, and piecewise log-linear odds ratio of lower severity, with change points at Day 0, 4, 10, 14, 16 and 26. The function call takes the form

```
model <- outcome(examination_time, clinical_status) ~ treatment_arm + baseline_severity
result2 <- po(formula = model,
              data = acttData,
              subject = 'subject_id',
              treatment = 'treatment_arm',
              imputation = T,
              knots = c(0, 4, 10, 14, 16, 26),
              start.time = 1, end.time = 28)
```

```
## [1] "start imputation"
## [1] "end imputation"
## [1] "start estimating common odds ratio"
## [1] "end estimating common odds ratio"
## [1] "start fitting piecewise linear model"
## [1] "end fitting piecewise linear model"
## [1] "start fitting separate proportional odds models"
## [1] "end fitting separate proportional odds models"
```

The estimated common odds ratio of lower severity and its 95% confidence interval are

```
result2$common.odds.ratio
```

```
## $estiamte
## treatment_arm
##      1.364902
##
## $confidence.interval
##                     2.5%    97.5%
## treatment_arm 1.033606 1.802386
```

The estimated daily odds ratios of lower severity and corresponding 95% confidence interval are

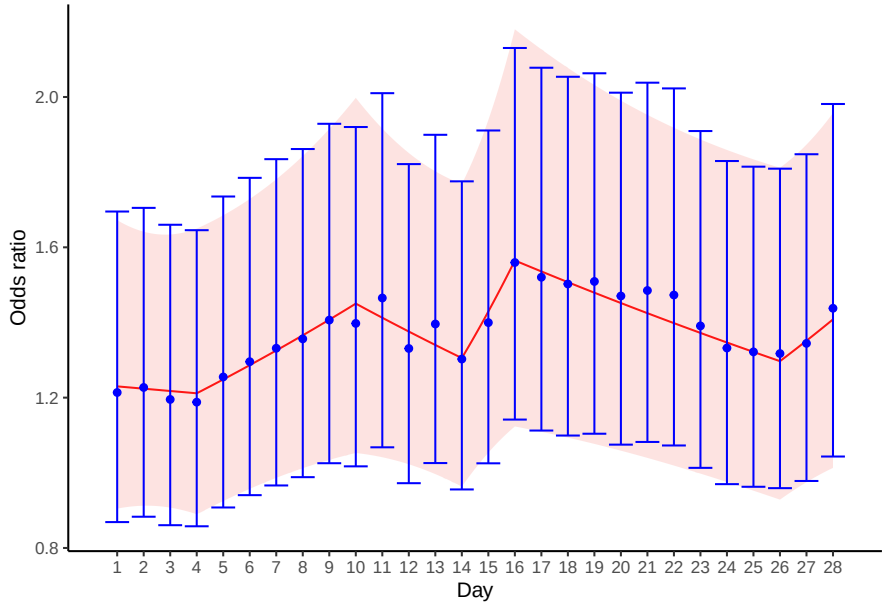
```
result2$separate.model
```

```
##    Time Odds Ratio      2.5%    97.5%
## 1     1   1.213688 0.8689097 1.695272
## 2     2   1.227090 0.8831608 1.704957
## 3     3   1.195244 0.8606031 1.660007
## 4     4   1.188031 0.8577170 1.645550
## 5     5   1.255043 0.9077457 1.735214
## 6     6   1.295668 0.9405605 1.784845
## 7     7   1.331481 0.9663521 1.834572
## 8     8   1.356453 0.9884177 1.861524
## 9     9   1.406371 1.0255074 1.928685
## 10   10   1.397530 1.0171818 1.920100
## 11   11   1.465133 1.0678805 2.010165
## 12   12   1.330829 0.9723738 1.821425
## 13   13   1.395948 1.0259460 1.899390
## 14   14   1.302649 0.9557122 1.775528
## 15   15   1.399696 1.0252805 1.910841
## 16   16   1.559551 1.1416645 2.130397
## 17   17   1.520373 1.1124726 2.077835
```

```
## 18   18   1.502501 1.0991822 2.053809
## 19   19   1.509173 1.1039727 2.063097
## 20   20   1.470456 1.0749495 2.011482
## 21   21   1.485073 1.0821245 2.038066
## 22   22   1.473068 1.0727300 2.022811
## 23   23   1.390887 1.0131977 1.909367
## 24   24   1.332091 0.9698690 1.829594
## 25   25   1.321802 0.9628554 1.814561
## 26   26   1.317330 0.9591704 1.809230
## 27   27   1.344441 0.9782671 1.847677
## 28   28   1.437788 1.0433450 1.981352
```

The comparison between the estimated piecewise log-linear odds ratios and the estimated daily odds ratios is provided by the following plot. The blue dots are daily odds ratios estimates under separate proportional odds models on each examination day, and the red curve shows the estimated piecewise log-linear odds ratios.

```
result2$piecewise.linear$plot
```



References

Lin, DY, Wang J., Gu Y., Zeng D (2022). Evaluating Treatment Efficacy in Hospitalized Covid-19 Patients. Submitted.

L. J. Wei, D. Y. Lin & L. Weissfeld (1989) Regression Analysis of Multivariate Incomplete Failure Time Data by Modeling Marginal Distributions, Journal of the American Statistical Association, 84:408, 1065-1073, DOI: 10.1080/01621459.1989.10478873