IntCens: Maximum Likelihood Estimation of Semiparametric Regression Models With Interval-Censored Data

1 Background

Interval-censored data arise when the failure time of each study subject is only known to lie in an interval. `IntCens` is an R software package that implements nonparametric maximum likelihood estimation (NPMLE) for a broad class of semiparametric regression models with general interval-censored data. The current release handles three types of failure time data:

- **Univariate.** Univariate failure time data described in Zeng, Mao and Lin (2016).
- **Multiple Events.** Failure times for multiple types of events.
- **Clustered.** Subjects are clustered, e.g. by family.

Users are encouraged to check the website https://dlin.web.unc.edu/software/IntCens/ for updates.

The IntCens package contains three executable programs:

`unireg`, `multireg`, and `clustreg`

Details on how/when to run these executables can be found below.

References:

Zeng, D., Mao, L., and Lin, D.Y. (2016). Maximum Likelihood Estimation for Semiparametric Transformation Models With Interval-Censored Data. Biometrika, in press.

2 Single Event Data

2.1 Data and Model

There are a total of n subjects in the study. For $i = 1, \ldots, n$, let T_i denote the event time of the ith subject, and $X_i(\cdot)$ denote a p-vector of possibly time-dependent covariates. The cumulative hazard function of T_i conditional on X_i takes the form

$$\Lambda(t; X_i) = G\left(\int_0^t e^{\beta^{\mathrm{T}} X_i(s)} d\Lambda(s)\right),$$

where β is a p-vector of regression parameters, $\Lambda(\cdot)$ is an arbitrary cumulative baseline hazard function, and $G(x) = r^{-1}\log(1 + rx)$ $(r \geq 0)$, with $G(x) = x$ under $r = 0$. We wish to estimate β and $\Lambda(\cdot)$.

2.2 Input Data Structure

There are two formats for the data file that can be used. The first should be used if any of the covariates are time-dependent; the second can be used for time-independent covariates. For both formats, the data file should have a single header line on the first line of the file which lists the column names.

2.2.1 Time-Dependent Covariates

The data file must contain the following columns: subject ID, examination time, status, and covariates. Each subject has a series of examination times. For each examination time, status indicates, by the values 1 versus 0, whether or not the event has occurred before that time, and covariates pertain to the measurements at that time. Additional columns are allowed, although not used. An example data file is given in Figure 1.

Subject_ID	Examination_Time	Status	Age	Gender	Needle	Jail	Inject	Race
030	3.73	0	28	MALE	0	1	0	WHITE
030	7.54	0	28	MALE	0	1	0	WHITE
031	3.16	0	33	FEMALE	0	0	1	OTHER
031	6.99	0	33	FEMALE	1	0	0	OTHER
031	13.55	1	33	FEMALE	0	1	0	OTHER
032	3.83	0	21	FEMALE	0	1	0	ASIAN
032	7.40	0	21	FEMALE	0	0	0	ASIAN
032	10.93	0	21	FEMALE	0	0	0	ASIAN
⋮								
068	3.87	1	30	MALE	1	1	1	OTHER
⋮								

Figure 1: Example Input Data File for Time-Dependent Covariates

In this study, Needle, Jail, and Inject are time-dependent covariates, whereas Age, Gender, and Race are time-independent covariates.

2.2.2 Time-Independent Covariates

Time-independent covariates can be treated as a special case of time-dependent covariates, such that it is possible to use the above input file format for time-independent covariates. Then, for a given subject, the covariate values will be the same among all examination times.

We also allow a simpler format for data with only time-independent covariates. In this format, each subject has only one record, which contains the left and right ends of the smallest time interval that brackets the failure time (i.e., the largest examination time that is smaller than the failure time and the smallest examination time that is larger than the failure time), as well as the value of each covariate. Figure 2 illustrates this format for the data used in Figure 1, where time-dependent covariates have been made time-independent by taking their values at the first examination time.

Left_Time	Right_Time	Age	Gender	Needle	Jail	Inject	Race
7.54	inf	28	MALE	0	1	0	WHITE
6.99	13.55	33	FEMALE	0	0	1	OTHER
10.93	inf	21	FEMALE	0	1	0	ASIAN
⋮							
0.0	3.87	30	MALE	1	1	1	OTHER
⋮							

Figure 2: Example Input Data File for Time-Independent Covariates

The symbol `inf` in Figure 2 denotes ∞. The `IntCens` package allows the user to specify different symbols for ∞ in the input file; see Section 2.5 for details.

Note that for both data formats, the column names should not include any mathematical symbols or spaces. For example, "Left-Time" and "Left Time" are invalid column names.

2.3 Usage

The program within the `IntCens` package that performs regression analysis of univariate failure time data is `unireg`. The format for `unireg` is as follows:

```
(Linux/Mac)     ./unireg --in PATH --out PATH --r r --model "MODEL" --subject_id ID
(Windows)   ./unireg.exe --in PATH --out PATH --r r --model "MODEL" --subject_id ID
```

The `--in PATH` argument specifies the path to the input data file, which has one of the two formats described above. The `--out PATH` argument specifies the path to the output file generated by `unireg`. The `r` argument is a numeric value in the class of logarithmic transformations:

$$G_r(x) = \begin{cases} x & r = 0 \\ \frac{\log(1+rx)}{r} & r > 0 \end{cases}$$

If `r` is not specified, the default is to use 0.0. The `subject_id` argument specifies the name of the column that contains the subject ID. This argument is present only for the input data format with time-dependent covariates.

The `model` argument has one of the following two formats, corresponding to the two input file formats:

Model Specification for Time-Dependent Covariates:

"(Examination_Time, Status) = $\texttt{Cov}_1$ + $\texttt{Cov}_2$ + ...+ $\texttt{Cov}_p$" (1)

The first term on the left-hand side of (1) refers to the examination time in the input data file (e.g., "Examination_Time" in Figure 1) and the second term refers to the status variable (e.g., "Status" in Figure 1).

Model Specification for Time-Independent Covariates:

"(Left_Time, Right_Time) = $\texttt{Cov}_1$ + $\texttt{Cov}_2$ + ...+ $\texttt{Cov}_p$" (2)

The first term on the left-hand side of (2) refers to the left end of the time interval that brackets the failure time (e.g., "Left_Time" in Figure 2) and the second term refers to the right end of the interval (e.g., "Right_Time" in Figure 2).

The right-hand sides of (1) and (2) can be any mathematical expression involving the covariates in the input data file. One example for the input file of Figure 1 is the following:

Needle + Needle^2 + Log(Age) + Gender + Race + Gender * Race

`unireg` automatically expands a categorical variable into a series of of indicator variables and creates numeric values for all non-numeric covariates.

2.4 Examples

Below are some examples of using `unireg` (all examples are for the Linux/Mac version; for Windows, simple change the name of the executable in the example commands below from `./unireg` to `./unireg.exe`):

Time-Dependent Covariates, r = 0.0:[1]

```
./unireg --in time_dep_cov.txt --out example_one_output.txt
        --model "(Examination_Time, Status) = Age + Gender + Needle + Jail + Inject"
        --subject_id Subject_ID
```

Time-Independent Covariates, r = 1.0:[2]

```
./unireg --in time_indep_cov.txt --out example_two_output.txt --r 1.0
        --model "(Left_Time, Right_Time) = Needle + Needle^2 + Log(Age) +
                 Gender + Race + Gender * Race"
```

[1] Running this example as-is on the given input file will fail: you need to include arguments:

```
--sep " " --max_itr 2000
```

These arguments will be explained in the next section.

[2] Running this example as-is on the given input file will fail: you need to include arguments:

```
--sep " " --max_itr 2000 --inf_char inf
```

These arguments will be explained in the next section.

2.5 Addtional Options

`unireg` offers additional options to allow greater flexibility in the input data file and perform certain data preprocessing tasks. The following (optional) arguments specify special characters used in the input data file:

- `--sep`: Specifies the character used to separate columns (column delimiter). The default is tab ('\t'). Note: If you get an 'ERROR in Parsing Model' that a variable does not appear on the header line, be sure you have specified the correct column delimiter.
- `--comment_char`: Lines beginning with this character are ignored.
- `--missing_value`: Specifies how missing values are represented. The default value is "NA".
- `--inf_char`: Specifies the special character(s) used to express infinite values. There is no default set, so if your data file has infinite values, you will need to use this argument to specify the character(s) used to represent infinity (e.g. "9999" or "inf").

The following two optional arguments can be used to adjust covariate values:

- `--extrapolation`
- `--collapse`

The `extrapolation` argument controls how to extrapolate values for time-dependent covariates between two examination times. The NPMLE requires the covariate values for each subject at all distinct left and right ends of the intervals that bracket the failure times, up through each subject's final examination time. If a subject is not examined at the baseline, then his/her covariate values at the first examination time are used automatically for all the time points before the first examination time. For a time point between two examination times for a given subject, there is a choice of using his/her covariate values at the nearest examination time on the left, the nearest examination time on the right, or the nearest examination time (left or right, whichever is closer). Another option is to treat a covariate as time-independent and just use the value at the first examination time. Choosing among these four alternatives is controlled by specifying one of four keywords: `nearest_left`, `nearest_right`, `nearest`, and `first`. The default value for `extrapolation` is `nearest_right`.

The `collapse` argument allows the user to collapse a set of original covariate values to a target value. For example, the user may want to convert an ordinal or count variable to a binary variable by setting all the values greater than 1 to 1. As another example, one may transform age (in years) into an orginal variable. The format for this option is:

`COVARIATE = {ORIG_VALUE(S)} -> TARGET_VALUE`

where `ORIG_VALUE(S)` can be a single string, numeric value, numeric range (format: `[a..b]`, with the special symbol "-inf" for `a` and "inf" for `b` to denote $\pm\infty$), or (comma-separated) list of these three; and `TARGET_VALUE` should be a single numeric or string value. (In the latter case, the user should use **single** quotes to emphasize a non-numeric value). The default is to not collapse any covariate.

If both options are applied to a given covariate, the order is to first apply `collapse` rules, and then `extrapolate`.

The following optional arguments can be used to fine-tune the algorithm used by `unireg`:

- `--convergence_threshold`
- `--max_itr`

`convergence_threshold` is the constant in the convergence criterion of the EM algorithm. The algorithm is deemed congerent when the maximum of the differences of the parameter estimates at two successive iterations is less than this constant. The default value is 0.0001. Use the `max_itr` argument to specify the maximum number of iterations before aborting the EM algorithm; the default is 1000.

2.5.1 Examples of Using the Optional Arguments

Time-Dependent Covariates with Additional Arguments:

```
./unireg --in time_dep_cov.txt --out example_three_output.txt
         --model "(Examination_Time, Status) = Age + Gender + Needle + Jail + Inject"
         --subject_id Subject_ID --missing_value 99 --max_itr 2000 --sep " "
         --extrapolation "Needle = first; Inject = first"
         --collapse "Gender={FEMALE}->0.0;Gender={MALE}->1.0;
                     Needle={[0.0..1.0]}->0.0;Needle={[2.0..inf]}->1.0;
                     Jail={[1.0..inf]}->1.0"
```

Time-Independent Covariates with Additional Arguments:

```
./unireg --in time_indep_cov.txt --out example_four_output.txt --r 1.0 --sep " "
         --model "(Left_Time, Right_Time) = Needle + Needle^2 + Log(Age) +
                  Gender + Race + Gender * Race"
         --collapse "Race={OTHER, ASIAN}->0.0;Race={WHITE}->1.0;
                     Age={[0..9]}->0;Age={[10..19]}->1;
                     Age={[20..29]}->2;Age={[30..39]}->3;
                     Age={[40..49]}->4;Age={[50..inf]}->5"
         --convergence_threshold 0.01 --inf_char inf
```

3 Multiple Events Data

3.1 Data and Model

There are a total of n subjects, each of whom may experience K types of events. For $i = 1, \ldots, n$ and $k = 1, \ldots, K$, let T_{ik} denote the kth event time of the ith subject, and $X_{ik}(\cdot)$ denote the corresponding p_k-vector of possibly time-dependent covariates. The cumulative hazard function of T_{ik} takes the form

$$\Lambda_{ik}(t) = G_k\left(\int_0^t e^{\beta_k^{\mathrm{T}} X_{ik}(s)+b_i} d\Lambda_k(s)\right),$$

where β_k is a p_k-vector of regression parameters, $\Lambda_k(\cdot)$ is an arbitrary cumulative baseline hazard function, b_i is a normal random variable with mean zero and variance σ^2, and $G_k(x) = r_k^{-1}\log(1+r_k x)$ $(r_k \geq 0)$, with $G_k(x) = x$ under $r_k = 0$. We wish to estimate $(\beta_1, \ldots, \beta_K)$, σ^2, and $(\Lambda_1, \ldots, \Lambda_K)$.

3.2 Input Data Structure

There is a column to indicate the event type. For time-independent covariates, each subject has K rows of observations that correspond to the K events. For time-dependent covariates, each subject has K blocks of observations, the kth block corresponding to the kth event (the number of rows per block, which corresponds to the number of observation times for that event and subject, need not be the same across subjects). For each event, the input data structure is the same as that of the single event data. Examples of each data file are in Figures 3 and 4 below, where there are two events *Diabt* and *Hypert*.

Subject_ID	Event	Time	Status	Center	Age	Sex	Race	BMI	Glucose	SysBP	DiaBP
67	Diabt	0	0	W	62	F	W	19.10	96.97	132	66
67	Diabt	1106	0	W	62	F	W	17.80	100.69	130	60
67	Diabt	2269	0	W	62	F	W	19.09	95.19	138	67
67	Diabt	3345	0	W	62	F	W	19.76	118.96	167	68
67	Hypert	0	0	W	62	F	W	19.10	96.97	132	66
67	Hypert	1106	0	W	62	F	W	17.80	100.69	130	60
67	Hypert	2269	0	W	62	F	W	19.09	95.19	138	67
67	Hypert	3345	1	W	62	F	W	19.76	118.96	167	68
139	Diabt	0	0	W	56	M	W	22.69	87.00	112	61
139	Hypert	0	0	W	56	M	W	22.69	87.00	112	61
⋮											

Figure 3: Example Input Data File for Multiple Events With Time-Dependent Covariates

Event	Left_Time	Right_Time	Center	Age	Sex	Race	BMI	Glucose	SysBP	DiaBP
Diabt	3345	Inf	W	62	F	W	19.10	96.97	132	66
Hypert	2269	3345	W	62	F	W	19.10	96.97	132	66
Diabt	0	Inf	W	56	M	W	22.69	87.00	112	61
Hypert	0	Inf	W	56	M	W	22.69	87.00	112	61
Diabt	8848	Inf	W	48	F	W	22.31	96.60	94	66
Hypert	8848	Inf	W	48	F	W	22.31	96.60	94	66
⋮										

Figure 4: Example Input Data File for Multiple Events With Time-Independent Covariates

3.3 Usage

The function within the `IntCens` package that performs regression analysis of multivariate failure time data is `multireg`. The format for `multireg` is as follows:

(Linux/Mac)
```
./multireg --in PATH --out PATH --r "r_1,r_2,...,r_k" --subject_id ID
           --model_1 "MODEL_1" --model_2 "MODEL_2" ...  --model_k "MODEL_k"
           --event_type TYPE --event_order "Event_1,Event_2,...,Event_k"
```
(Windows)
```
./multireg.exe --in PATH --out PATH --r "r_1,r_2,...,r_k" --subject_id ID
               --model_1 "MODEL_1" --model_2 "MODEL_2" ...  --model_k "MODEL_k"
               --event_type TYPE --event_order "Event_1,Event_2,...,Event_k"
```

The `--in`, `--out`, and `--subject_id` (which should be specified if and only if the data is in time-dependent format) arguments are the same as for `unireg`.

The `--r` argument is a comma-separated list of numeric values, corresponding to the r-value to use for each event. If `--r` is not specified, the default is to use 0.0 for every event. Note that, in contrast to the `unireg` usage, the `--r` argument should be enclosed in quotes, since it is no longer a numeric value (but rather a list of values).

The `--event_type` argument specifies the name of the column that contains the event type information. The `--event_order` argument is a comma-separated list of all the event types. Note that this list determines the order of the events, so that the k^{th} item in this list corresponds to `--model_k`, and similarly the kth r-value will be applied to the kth event type.

The `--model_i` argument(s) specify the models, where the format of each model is the same as was described for the `--model` argument for `unireg`. If only one model is provided, the same model will be used for all events; otherwise specify the model for the ith event via `--model_i`.

The same additional arguments mentioned for `unireg` in Section 2.5 can also be used for `multireg`.

3.4 Examples

Below are some examples of using `multireg` (all examples are for the Linux/Mac version; for Windows, simple change the name of the executable in the example commands below from `./multireg` to `./multireg.exe`):

Time-Dependent Covariates, r = (0.0,0.0):

```
./multireg --in MultType_Time_Dependent.txt --out example_five_output.txt --sep " "
           --model "(Time,Status)=Center+Age+Sex+Race+BMI+Glucose+SysBP+DiaBP"
           --r "0.0,0.0" --subject_id Subject_ID --inf_char Inf
           --event_type Event --event_order "Diabt,Hypert"
```

Time-Independent Covariates, r = (1.0,1.0):

```
./multireg --in MultType_Time_Independent.txt --out example_six_output.txt --sep " "
           --model_1 "(Left_Time,Right_Time)=Center+Age+Sex+Race+BMI+Glucose"
           --model_2 "(Left_Time,Right_Time)=Center+Age+Sex+Race+BMI+SysBP+DiaBP"
           --r "0.0,0.0" --event_type Event --event_order "Diabt,Hypert" --inf_char Inf
```

4 Clustered Data

4.1 Data and Model

There are n clusters, with n_i subjects in the ith cluster. For $i = 1, \ldots, n$ and $j = 1, \ldots, n_i$, let T_{ij} denote the event time for the jth subject of the ith cluster, and $X_{ij}(\cdot)$ denote the corresponding p-vector of possibly time-dependent covariates. The cumulative hazard function of T_{ij} takes the form

$$\Lambda_{ij}(t) = G\left(\int_0^t e^{\beta^{\mathrm{T}} X_{ij}(s)+b_i} d\Lambda(s)\right),$$

where β is a p-vector of regression parameters, $\Lambda(\cdot)$ is an arbitrary cumulative baseline hazard function, b_i is a normal random variable with mean zero and variance σ^2, and $G(x) = r^{-1}\log(1 + rx)$ $(r \geq 0)$, with $G(x) = x$ under $r = 0$. We wish to estimate β, σ^2, and $\Lambda(\cdot)$.

4.2 Input Data Structure

There is a column to denote the cluster ID. For time-independent covariates, the ith cluster contributes n_i rows of observations that correspond to the n_i subjects of the ith cluster. For time-dependent covariates, the ith cluster contributes n_i blocks of observations, the jth block pertaining to the jth subject of the ith cluster. For each subject, the data structure is the same as that of the single event data. Examples of each data file are in Figures 5 and 6 below.

Cluster_ID	**Subject_ID**	**Time**	**Status**	**Center**	**Age**	**Sex**	**Race**	**BMI**	**Glucose**	**SysBP**	**DiaBP**
1	267	0	0	W	52	F	W	19.74	85.51	107	64
1	267	1071	0	W	52	F	W	19.95	93.07	96	63
1	919	0	0	W	52	M	W	25.92	94.49	99	61
1	919	1006	0	W	52	M	W	26.10	109.43	93	61
1	919	2110	0	W	52	M	W	26.92	114.61	97	68
1	919	3225	0	W	52	M	W	25.91	102.52	91	59
1	919	8857	0	W	52	M	W	30.24	121.89	117	73
2	1434	0	0	W	65	M	W	26.19	90.07	123	74
2	1434	981	0	W	65	M	W	23.88	98.34	117	65
2	1434	2104	0	W	65	M	W	24.44	93.81	134	63
2	1434	3233	0	W	65	M	W	23.89	93.12	125	56
⋮											

Figure 5: Example Input Data File for Clustered Data With Time-Dependent Covariates

Cluster_ID	Left_Time	Right_Time	Center	Age	Sex	Race	BMI	Glucose	SysBP	DiaBP
1	1071	Inf	W	52	F	W	19.74	19.74	107	64
1	8857	Inf	W	52	M	W	25.92	25.92	99	61
2	3233	Inf	W	65	M	W	26.19	26.19	123	74
3	1094	Inf	W	63	M	B	21.41	21.41	114	86
3	1045	Inf	W	50	M	W	24.88	24.88	114	72
3	3131	8488	W	49	M	W	25.62	25.62	127	91
3	8083	Inf	W	48	F	W	29.82	29.82	111	70
⋮										

Figure 6: Example Input Data File for Clustered Data With Time-Independent Covariates

4.3 Usage

The function within the `IntCens` package that performs regression analysis of clustered failure time data is `clustreg`. The format for `clustreg` is as follows:

(Linux/Mac) `./clustreg --in PATH --out PATH --r r --model "MODEL" --cluster_id ID`

(Windows) `./clustreg.exe --in PATH --out PATH --r r --model "MODEL" --cluster_id ID`

The `input`, `output`, `r`, `model`, and `subject_id` arguments are all the same as for `unireg`. The only additional argument is the `cluster_id`, which specifies the name of the column that contains the cluster id.

The same additional arguments mentioned for `unireg` in Section 2.5 can also be used for `clustreg`.

4.4 Examples

Below are some examples of using `clustreg` (all examples are for the Linux/Mac version; for Windows, simple change the name of the executable in the example commands below from `./clustreg` to `./clustreg.exe`):

Time-Dependent Covariates, r = 0.0:

```
./clustreg --in Cluster_Time_Dependent.txt --out example_seven_output.txt --r 0.0
           --model "(Time,Status)=Center+Age+Sex+Race+BMI+Glucose+SysBP+DiaBP"
           --subject_id Subject_ID --cluster_id Cluster_ID --inf_char Inf --sep " "
```

Time-Independent Covariates, r = 1.0:

```
./clustreg --in Cluster_Time_Independent.txt --out example_eight_output.txt --r 1.0
           --model "(Left_Time,Right_Time)=Center+Age+Sex+Race+BMI+Glucose+SysBP+DiaBP"
           --cluster_id Cluster_ID --inf_char Inf --sep " "
```