

PROGRAM LISTINGS FOR SAS AND STATA

Here is the program code using either SAS or Stata for all the analyses described in *Event History and Survival Analysis* (Second Edition) by Paul D. Allison, published by Sage Publications in 2014. Enquiries should be directed to allison@soc.upenn.edu. Data sets can be downloaded at www.statisticalhorizons.com/resources/data-sets.

Table 2.2

Stata

**Convert data from wide form to long form*

```
use c:\data\rank.dta, clear
gen id=_n
reshape long art cit, i(id) j(year)
drop if year>dur
replace promo=0 if year<dur
gen jobpres=prest1
replace jobpres = prest2 if year>=jobtime
```

**Estimate models*

```
logit promo undgrad phdmed phdprest jobpres art cit
logit, or
logit promo undgrad phdmed phdprest jobpres art cit year c.year#c.year
logit, or
logit promo undgrad phdmed phdprest jobpres art cit year c.year#c.year
      c.phdprest#c.year
```

SAS

**Convert data from wide form to long form;*

```
data my.rankyrs;
  set my.rank;
  array arts(*) art1-art10;
  array cits(*) cit1-cit10;
  if jobtime=. then jobtime=11;
  do year=1 to dur;
    if year=dur then promo=event; else promo=0;
    if year ge jobtime then jobpres=prest2; else jobpres=prest1;
    art=arts(year);
    cit=cits(year);
    output;
  end;
run;
```

**Estimate models;*

```
proc logistic data=my.rankyrs;
  model promo(desc)=undgrad phdmed phdprest art cit jobpres;
proc logistic data=my.rankyrs;
  model promo(desc)=undgrad phdmed phdprest art cit jobpres
```

```

    year year*year;
proc logistic data=my.rankyrs;
    model promo(desc)=undgrad phdmed phdprest art cit jobpres
        year year*year phdprest*year;
run;

```

Table 2.3

Stata

**Scenario A*

```

use c:\data\rank.dta, clear
gen id=_n
reshape long art cit, i(id) j(year)
drop if year>dur
replace promo=0 if year<dur
replace promo=1 if year==dur & dur<10
gen jobpres=prest1
replace jobpres = prest2 if year>=jobtime
logistic promo undgrad phdmed phdprest jobpres art cit year c.year#c.year

```

**Scenario B*

```

use c:\data\rank.dta, clear
gen id=_n
reshape long art cit, i(id) j(year)
drop if year>dur & promo==1
replace promo=0 if year<dur
replace art=art[_n-1] if art==.
replace cit=cit[_n-1] if cit==.
gen jobpres=prest1
replace jobpres = prest2 if year>=jobtime
logistic promo undgrad phdmed phdprest jobpres art cit year c.year#c.year

```

SAS

**Scenario A;*

```

data rankyrs;
set my.rank;
    array arts(*) art1-art10;
    array cits(*) cit1-cit10;
    if jobtime=. then jobtime=11;
    promo=0;
    do year=1 to dur;
        if year=dur and dur<10 then promo=1;
        if year=dur and dur=10 then promo=event;
        if year ge jobtime then jobpres=prest2;
            else jobpres=prest1;
        art=arts(year);
        cit=cits(year);
        output;
    end;

```

```

run;
proc logistic data=rankyrs;
  model promo(desc)=undgrad phdmed phdprest art cit jobpres
    year year*year;
run;

*Scenario B;
data rankyrs;
set my.rank;
  array arts(*) art1-art10;
  array cits(*) cit1-cit10;
  if jobtime=. then jobtime=11;
  if event=1 then do year=1 to dur;
    if year=dur then promo=1; else promo=0;
    if year ge jobtime then jobpres=prest2;
    else jobpres=prest1;
    art=arts(year);
    cit=cits(year);
    output;
  end;
  if event=0 then do year=1 to 10;
    promo=0;
    if year ge jobtime then jobpres=prest2;
    else jobpres=prest1;
    art=arts(year);
    if art=. then art=arts(dur);
    cit=cits(year);
    if cit=. then cit=cits(dur);
    output;
  end;
proc logistic data=rankyrs;
  model promo(desc)=undgrad phdmed phdprest art cit jobpres
    year year*year;
run;

```

Table 3.1 , Figures 3.1- 3.2

Stata

```

use c:\data\recid.dta, clear
stset week, failure(arrest==1)
streg fin age race wexp mar paro prio, dist(exponential)
streg, nohr
streg fin age race wexp mar paro prio, dist(weibull)
streg, nohr
stcurve, hazard
streg fin age race wexp mar paro prio, dist(gamma)
streg, tr
streg fin age race wexp mar paro prio, dist(llogistic)
stcurve, hazard

```

SAS

```
proc lifereg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio / d=exponential;
proc lifereg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio / d=weibull;
proc lifereg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio / d=gamma;
run;
```

Tables 4.1-4.4

Stata

```
use c:\data\recid.dta, clear
stset week, failure(arrest==1)
stcox fin age race wexp mar paro prio
stcox, nohr
gen id=_n
reshape long work, i(id) j(stop)
gen start=stop-1
drop if stop > week
replace arrest=0 if week~=stop
stset stop, failure(arrest==1) id(id) time0(start)
stcox fin age race wexp mar paro prio work
stcox, nohr
estat phtest, detail
generate worklag=work[_n-1] if start>0
stcox fin age race wexp mar paro prio worklag
stcox, nohr
stcox fin age race wexp mar paro prio work, tvc(age wexp)
stcox, nohr
stcox fin age race mar paro prio work, strata(wexp)
stcox, nohr
```

SAS

```
proc phreg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio;
proc phreg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio work;
  array wrk(*) w1-w52;
  work = wrk[week];
proc phreg data=my.recid;
  where week ne 1;
  model week*arrest(0)=fin age race wexp mar paro prio work;
  array wrk(*) w1-w52;
  work = wrk[week-1];
proc phreg data=my.recid;
  model week*arrest(0)=fin age race wexp mar paro prio work ageweek
```

```

    wexpweek;
    array wrk(52) w1-w52;
    work = wrk[week];
    agetime=age*week;
    wexptime=wexp*week;
proc phreg data=my.recid;
    model week*arrest(0)=fin age race mar paro prio work;
    array wrk(*) w1-w52;
    work = wrk[week];
    strata wexp;
run;

```

Table 4.6

Stata

```

use c:\data\recid.dta, clear
stset week, failure(arrest==1)
stcox fin age race wexp mar paro prio
stcurve, survival at(fin=1 age=21 race=1 wexp=1 mar=0 paro=1 prio=4)
    outfile(surv,replace)

```

SAS

```

data cov;
    input fin age race wexp mar paro prio;
    datalines;
1 21 1 1 0 1 4
proc phreg data=my.recid;
    model week*arrest(0)=fin age race wexp mar paro prio;
    baseline out=surv covariates=cov survival=s upper=u lower=l;
proc print data=surv;
    var s l u;
run;

```

Table 5.1

Stata

```

use c:\data\tarp.dta, clear
stset arrstday, failure(type==1 2)
stcox fin age white male married paro numprop crimprop numarst edcomb
stcox, nohr
stset arrstday, failure(type==1)
stcox fin age white male married paro numprop crimprop numarst edcomb
stcox, nohr
stset arrstday, failure(type==2)
stcox fin age white male married paro numprop crimprop numarst edcomb
stcox, nohr

```

SAS

```
proc phreg data=my.tarp;
model arrstday*type(0)=fin age white male married paro
  numprop crimprop numarst edcomb;
proc phreg data=my.tarp;
model arrstday*type(0 1)=fin age white male married paro
  numprop crimprop numarst edcomb;
proc phreg data=my.tarp;
model arrstday*type(0 2)=fin age white male married paro
  numprop crimprop numarst edcomb;
run;
```

Figure 5.1

Stata

```
use c:\data\tarp.dta, clear
stset arrstday, failure(type==1)
stcompet CumInc=ci, compet1(2)
gen CumInc1 = CumInc if type==1
gen CumInc2 = CumInc if type==2
line CumInc1 CumInc2 _t, c(J J) sort
```

Table 5.2, Figure 5.2

Stata

```
use c:\data\tarp.dta, clear
stset arrstday, failure(type==2)
stcrreg fin age white male married paro numprop crimprop numarst edcomb,
  compete(type==1)
stcrreg, noshr
stset arrstday, failure(type==1)
stcrreg fin age white male married paro numprop crimprop numarst edcomb,
  compete(type==2)
stcrreg, noshr
stcurve, cif
```

Table 6.2

Stata

```
use c:\data\tarp.dta, clear
nbreg arrstcount fin age white male married paro numprop crimprop numarst
  edcomb
nbreg, irr
use c:\data\arrests.dta, clear
stset length, failure(arrind==1)
stcox fin age white male married paro numprop crimprop numarst edcomb
stcox, nohr
stcox fin age white male married paro numprop crimprop numarst edcomb,
  cluster(id)
```

```

stcox, nohr
set mat 1000
stcox fin age white male married paro numprop crimprop numarst edcomb,
    shared(id)
stcox, nohr

```

SAS

```

proc genmod data=my.tarp;
    model arrstcount=fin age white male married paro
        numprop crimprop numarst edcomb /d=nb ;
proc phreg data=my.arrests;
    model length*arrind(0)= fin age white male married paro
        numprop crimprop numarst edcomb;
proc phreg data=my.arrests covs(aggregate);
    model length*arrind(0)= fin age white male married paro
        numprop crimprop numarst edcomb;
    id id;
proc phreg data=my.arrests;
    class id;
    model length*arrind(0)= fin age white male married paro
        numprop crimprop numarst edcomb;
    random id;
run;

```

Table 6.3

Stata

```

use c:\data\arrests.dta, clear
stset length, failure(arrind==1)
streg spellnum fin age white male married paro numprop crimprop numarst
    edcomb, cluster(id) dist(weibull)
streg, nohr
streg spellnum fin age white male married paro numprop crimprop numarst
    edcomb, shared(id) dist(weibull)
streg, nohr
stset end, failure(arrind==1) time0(begin)
stcox fin age white male married paro numprop crimprop numarst edcomb,
    cluster(id)
stcox, nohr
stcox fin age white male married paro numprop crimprop numarst edcomb,
    cluster(id) tvc(numarst) texp(_t/30.4)

```

SAS

Weibull not available with robust standard errors or shared frailty.

```

proc phreg data=my.arrests covs(aggregate);
    model (begin,end)*arrind(0)= fin age white male married paro
        numprop crimprop numarst edcomb;
    id id;
proc phreg data=my.arrests covs(aggregate);

```

```
model (begin,end)*arrind(0)= fin age white male married paro
      numprop crimprop numarst edcomb numtime;
      numtime=numarst*(end/30.4);
id id;
run;
```