

BIO 223 Applied Survival Analysis Chapter 7: Time-varying covariates

References

- Cox Proportional-Hazards Regression for Survival Data: <http://cran.r-project.org/doc/contrib/Fox-Companion/appendix-cox-regression.pdf>
- BIO 223 Applied Survival Analysis note chapter 7.
- Time-dependent covariates in the Cox proportional-hazards regression model.: <http://www.ncbi.nlm.nih.gov/pubmed/10352854>

Recidivism example: Create dataset

“The file Rossi.txt contains data from an experimental study of recidivism of 432 male prisoners, who were observed for a year after being released from prison (Rossi, Berk, and Lenihan, 1980).”

Use of time-varying variables require a counting process (long format) dataset in R. Thus, wide-to-long transformation is done below.

```
## Load survival package
library(survival)

## Load data from online
Rossi <- read.table(file = "http://cran.r-project.org/doc/contrib/Fox-Companion/Rossi.txt", header = TRUE)
head(Rossi)
```

[illegible]

```
## Change to long format
Rossi.long <- reshape(data      = Rossi,
                      varying   = paste0("emp", 1:52),
                      v.names   = "employed",
                      timevar   = "time",
                      idvar     = "id",
                      direction = "long",
                      sep       = ""
                      )

## Sort by id and time
library(doby)
Rossi.long <- orderBy( ~  + id + time, data = Rossi.long)

## Drop rows where emp is NA (time after event/censoring)
Rossi.long <- Rossi.long[!is.na(Rossi.long$emp),]

## Load plyr package
library(plyr)
library(doMC)           # parallel backend to foreach/plyr
registerDoMC()          # Turn on multicore processing

## Create time variables and various forms of exposure variables
Rossi.long <- ddply(.data = Rossi.long,
                   .variables = c("id"),
                   .drop = TRUE, .parallel = TRUE,
                   .fun = function(DF) {

                       ## Start time is the start of each interval
                       DF$start <- c(0, head(DF$time, -1))

                       ## Stop time is the end of each interval
                       DF$stop <- DF$time

                       ## Event indicator for each interval
                       DF$event <- 0
                       ## Use arrest value for the last interval
                       DF[nrow(DF),"event"] <- DF[nrow(DF),"arrest"]

                       ## Initial employment status
                       DF$employed.initial <- DF$employed[1]

                       ## Lagged employment status (employment status from last interval matters)
                       DF$employed.lag1 <- c(rep(NA, 1), head(DF$employed, -1))

                       ## Cumulative number of weeks in employment
                       DF$employed.cumsum <- cumsum(DF$employed)

                       ## Ever employed status
                       DF$employed.ever <- as.numeric(DF$employed.cumsum > 0)

                       ## % of time in employment
                       DF$employed.percent <- with(DF, employed.cumsum / stop)*100

                       ## Return DF
                       DF
                   })
```

Functional forms of exposure

```
## Data for subject 2
Rossi.long[Rossi.long$id == 2,
c("id","start","stop","event","employed","employed.initial","employed.lag1","employed.ever","employed.cumsum","employed.percent")]
```

	id	start	stop	event	employed	employed.initial	employed.lag1	employed.ever	employed.cumsum	employed.percent
21	2	0	1	0	0	0	NA	0	0	0.00
22	2	1	2	0	0	0	0	0	0	0.00
23	2	2	3	0	0	0	0	0	0	0.00
24	2	3	4	0	0	0	0	0	0	0.00
25	2	4	5	0	0	0	0	0	0	0.00
26	2	5	6	0	0	0	0	0	0	0.00
27	2	6	7	0	0	0	0	0	0	0.00
28	2	7	8	0	0	0	0	0	0	0.00
29	2	8	9	0	0	0	0	0	0	0.00
30	2	9	10	0	1	0	0	1	1	10.00
31	2	10	11	0	1	0	1	1	2	18.18
32	2	11	12	0	1	0	1	1	3	25.00
33	2	12	13	0	1	0	1	1	4	30.77
34	2	13	14	0	1	0	1	1	5	35.71
35	2	14	15	0	0	0	1	1	5	33.33
36	2	15	16	0	0	0	0	1	5	31.25
37	2	16	17	1	0	0	0	1	5	29.41

The employment status variable has to be modeled in the “right” functional form to make correct inference about the exposure-outcome relationship. There are many different ways to model the exposure, and choosing the “right” functional form requires scientific understanding of the exposure-outcome relationship, not just statistics.

Here we created the current employment status (employed), the initial employment status (employed.initial), the past-week employment status (employed.lag1), ever-employed status (employed.ever), cumulative sum of weeks in employment (employed.cumsum), and % of time after release spent in employment (employed.percent).

Use cluster(id) to indicate the rows with the same id are from the same individual, thus, clustered. This will give the robust SE. However, the same analysis in SAS appear to return the SE with using robust estimator by default.

Using initial employment status (time-invariant)

Employment during the first week is associated with a HR for arrest of 0.7396 (non-significant). This is an equivalent of intention-to-treat (ITT) analysis.

```
res.with.employed.initial <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed.initial
+ cluster(id),
        data      = Rossi.long,
        ties      = c("efron","breslow","exact")[1])
summary(res.with.employed.initial)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed.initial + cluster(id),
      data = Rossi.long, ties = c("efron", "breslow", "exact")[1])

n= 19809, number of events= 114

              coef exp(coef) se(coef) robust se      z Pr(>|z|)
fin             -0.3801   0.6838   0.1913   0.1955 -1.94  0.0519 .
age             -0.0568   0.9448   0.0220   0.0254 -2.23  0.0257 *
race              0.2849   1.3296   0.3095   0.2949  0.97  0.3341
wexp            -0.1371   0.8719   0.2125   0.2188 -0.63  0.5308
mar             -0.4161   0.6596   0.3815   0.3785 -1.10  0.2717
paro            -0.0766   0.9263   0.1957   0.1994 -0.38  0.7009
prio              0.0905   1.0947   0.0286   0.0290  3.12  0.0018 **
employed.initial -0.3017   0.7396   0.3211   0.3131 -0.96  0.3352
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
fin              0.684      1.462      0.466      1.003
age              0.945      1.058      0.899      0.993
race             1.330      0.752      0.746      2.370
wexp             0.872      1.147      0.568      1.339
mar              0.660      1.516      0.314      1.385
paro             0.926      1.080      0.627      1.369
prio             1.095      0.913      1.034      1.159
employed.initial 0.740      1.352      0.400      1.366

Concordance= 0.643 (se = 0.027 )
Rsquare= 0.002 (max possible= 0.066 )
Likelihood ratio test= 34.2 on 8 df,  p=0.0000371
wald test               = 31.6 on 8 df,  p=0.000109
Score (logrank) test = 34.3 on 8 df,  p=0.0000359,   Robust = 27.9  p=0.000488

 (Note: the likelihood ratio and score tests assume independence of
        observations within a cluster, the wald and robust score tests do not).
```

Using time-dependent current employment status

Employment during a week is associated with a HR for arrest of 0.2649 during that week. The assumption is that only the current employment status matters, not past history of employment.

```
res.with.employed <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed +
cluster(id),
        data      = Rossi.long,
        ties      = c("efron","breslow","exact")[1])
summary(res.with.employed)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed + cluster(id), data = Rossi.long,
      ties = c("efron", "breslow", "exact"))[1])

n= 19809, number of events= 114

      coef exp(coef) se(coef) robust se      z    Pr(>|z|)
fin    -0.3567   0.7000  0.1911   0.1961 -1.82   0.0689 .
age    -0.0463   0.9547  0.0217   0.0251 -1.84   0.0653 .
race     0.3387   1.4031  0.3096   0.2978  1.14   0.2554
wexp    -0.0256   0.9748  0.2114   0.2194 -0.12   0.9073
mar     -0.2937   0.7455  0.3830   0.3975 -0.74   0.4599
paro    -0.0642   0.9378  0.1947   0.1992 -0.32   0.7472
prio     0.0851   1.0889  0.0290   0.0293  2.91   0.0036 **
employed -1.3283   0.2649  0.2507   0.2515 -5.28 0.00000013 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
fin           0.700      1.429    0.477    1.028
age           0.955      1.047    0.909    1.003
race          1.403      0.713    0.783    2.515
wexp          0.975      1.026    0.634    1.498
mar           0.745      1.341    0.342    1.625
paro          0.938      1.066    0.635    1.386
prio          1.089      0.918    1.028    1.153
employed      0.265      3.775    0.162    0.434

Concordance= 0.708 (se = 0.027 )
Rsquare= 0.003 (max possible= 0.066 )
Likelihood ratio test= 68.7 on 8 df, p=9.11e-12
Wald test = 57.3 on 8 df, p=0.00000000159
Score (logrank) test = 64.5 on 8 df, p=6.1e-11, Robust = 54.1 p=0.00000000647

(Note: the likelihood ratio and score tests assume independence of
      observations within a cluster, the Wald and robust score tests do not).
```

Using time-dependent past-week employment status (employment status of the previous interval is used)

Employment during a week is associated with a HR for arrest of 0.4553 during the next week. The assumption is that only the past-week employment status matters, not the current employment status or that of two or more weeks before.

```
res.with.employed.lag1 <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed.lag1 +
        cluster(id),
        data      = Rossi.long,
        ties      = c("efron","breslow","exact"))[1])
summary(res.with.employed.lag1)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed.lag1 + cluster(id), data = Rossi.long,
      ties = c("efron", "breslow", "exact")[1])

n= 19377, number of events= 113
(432 observations deleted due to missingness)

              coef exp(coef) se(coef) robust se      z Pr(>|z|)
fin          -0.3513   0.7038  0.1918   0.1957 -1.80  0.07259 .
age          -0.0498   0.9514  0.0219   0.0252 -1.98  0.04802 *
race           0.3215   1.3792  0.3091   0.2946  1.09  0.27521
wexp         -0.0476   0.9535  0.2132   0.2185 -0.22  0.82741
mar          -0.3448   0.7084  0.3832   0.3922 -0.88  0.37941
paro         -0.0471   0.9540  0.1963   0.1987 -0.24  0.81260
prio           0.0920   1.0964  0.0288   0.0286  3.22  0.00128 **
employed.lag1 -0.7869   0.4553  0.2181   0.2173 -3.62  0.00029 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
fin              0.704      1.421    0.480    1.033
age              0.951      1.051    0.906    1.000
race             1.379      0.725    0.774    2.457
wexp             0.953      1.049    0.621    1.463
mar              0.708      1.412    0.328    1.528
paro             0.954      1.048    0.646    1.408
prio             1.096      0.912    1.037    1.159
employed.lag1    0.455      2.197    0.297    0.697

Concordance= 0.67 (se = 0.027 )
Rsquare= 0.002 (max possible= 0.067 )
Likelihood ratio test= 47.2 on 8 df, p=0.000000143
Wald test = 41.6 on 8 df, p=0.00000158
Score (logrank) test = 46.4 on 8 df, p=0.000000199, Robust = 36.8 p=0.0000125

(Note: the likelihood ratio and score tests assume independence of
      observations within a cluster, the Wald and robust score tests do not).
```

Using time-dependent ever-employed status

Having experienced any duration of employment after release is associated with a HR for arrest of 0.6464. The assumption is that having experienced employment is equal regardless of duration or continuity.

```
res.with.employed.ever <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed.ever +
        cluster(id),
        data = Rossi.long,
        ties = c("efron","breslow","exact")[1])
summary(res.with.employed.ever)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed.ever + cluster(id), data = Rossi.long,
      ties = c("efron", "breslow", "exact")[1])

n= 19809, number of events= 114

      coef exp(coef) se(coef) robust se      z Pr(>|z|)
fin      -0.3773    0.6857  0.1913    0.1950 -1.93  0.0531 .
age      -0.0583    0.9434  0.0220    0.0254 -2.30  0.0217 *
race       0.3192    1.3761  0.3078    0.2890  1.10  0.2693
wexp      -0.0929    0.9113  0.2143    0.2164 -0.43  0.6678
mar       -0.3825    0.6821  0.3819    0.3807 -1.00  0.3149
paro      -0.0767    0.9261  0.1958    0.1990 -0.39  0.6997
prio       0.0878    1.0917  0.0288    0.0287  3.06  0.0022 **
employed.ever -0.4364    0.6464  0.2168    0.2129 -2.05  0.0404 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
fin           0.686      1.458    0.468    1.005
age           0.943      1.060    0.898    0.991
race          1.376      0.727    0.781    2.425
wexp          0.911      1.097    0.596    1.393
mar           0.682      1.466    0.323    1.438
paro          0.926      1.080    0.627    1.368
prio          1.092      0.916    1.032    1.155
employed.ever 0.646      1.547    0.426    0.981

Concordance= 0.652 (se = 0.027 )
Rsquare= 0.002 (max possible= 0.066 )
Likelihood ratio test= 37.1 on 8 df, p=0.0000108
Wald test = 37.7 on 8 df, p=0.00000862
Score (logrank) test = 37.5 on 8 df, p=0.00000926, Robust = 31.2 p=0.000127

(Note: the likelihood ratio and score tests assume independence of
      observations within a cluster, the Wald and robust score tests do not).
```

Using cumulative time in employment (total weeks of employment)

One additional week of employment is associated with a HR for arrest of 0.9825 regardless of when it is from (non-significant). The effect of the cumulative exposure for a given individual can increase (during additional exposure) or stay the same (while unexposed). The assumption is that each additional week of employment is equivalent no matter when it was obtained and that once obtained the benefit of employment will not degrade even if the subject becomes unemployed.

```
res.with.employed.cumsum <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed.cumsum +
        cluster(id),
        data    = Rossi.long,
        ties    = c("efron","breslow","exact")[1])
summary(res.with.employed.cumsum)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed.cumsum + cluster(id),
      data = Rossi.long, ties = c("efron", "breslow", "exact"))[1])

n= 19809, number of events= 114

      coef exp(coef) se(coef) robust se      z Pr(>|z|)
fin      -0.37962   0.68412  0.19121   0.19562 -1.94  0.0523 .
age      -0.05294   0.94844  0.02212   0.02565 -2.06  0.0390 *
race       0.31168   1.36572  0.30848   0.29445  1.06  0.2898
wexp      -0.08529   0.91825  0.21445   0.22215 -0.38  0.7010
mar       -0.37944   0.68425  0.38307   0.38253 -0.99  0.3212
paro      -0.06339   0.93858  0.19571   0.19997 -0.32  0.7512
prio       0.08907   1.09315  0.02887   0.02920  3.05  0.0023 **
employed.cumsum -0.01766   0.98250  0.00947   0.00964 -1.83  0.0671 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
fin           0.684      1.462     0.466     1.004
age           0.948      1.054     0.902     0.997
race          1.366      0.732     0.767     2.432
wexp          0.918      1.089     0.594     1.419
mar           0.684      1.461     0.323     1.448
paro          0.939      1.065     0.634     1.389
prio          1.093      0.915     1.032     1.158
employed.cumsum 0.982      1.018     0.964     1.001

Concordance= 0.649 (se = 0.027 )
Rsquare= 0.002 (max possible= 0.066 )
Likelihood ratio test= 36.9 on 8 df, p=0.0000121
Wald test = 34 on 8 df, p=0.0000414
Score (logrank) test = 36.5 on 8 df, p=0.000014, Robust = 30.2 p=0.000195

(Note: the likelihood ratio and score tests assume independence of
observations within a cluster, the Wald and robust score tests do not).
```

Using cumulative time in employment as percentage of time after release

One additional percent of time after release in employment is associated with a HR for arrest of 0.9923. The effect of the cumulative exposure for a given individual can increase (during additional exposure) or decrease (while unexposed). The assumption is that more time spent in employment is beneficial, but the benefit will degrade if the subject becomes unemployed.

```
res.with.employed.percent <-
  coxph(formula = Surv(start, stop, event) ~ fin + age + race + wexp + mar + paro + prio + employed.percent
+ cluster(id),
      data = Rossi.long,
      ties = c("efron","breslow","exact"))[1])
summary(res.with.employed.percent)
```

```
Call:
coxph(formula = Surv(start, stop, event) ~ fin + age + race +
      wexp + mar + paro + prio + employed.percent + cluster(id),
      data = Rossi.long, ties = c("efron", "breslow", "exact"))[1])

n= 19809, number of events= 114

      coef exp(coef) se(coef) robust se      z Pr(>|z|)
fin    -0.38011   0.68379  0.19116   0.19575 -1.94   0.052  .
age    -0.05121   0.95008  0.02216   0.02591 -1.98   0.048  *
race     0.30876   1.36174  0.30867   0.29419  1.05   0.294
wexp    -0.06226   0.93964  0.21415   0.22093 -0.28   0.778
mar     -0.34864   0.70565  0.38341   0.38707 -0.90   0.368
paro    -0.05265   0.94872  0.19576   0.20052 -0.26   0.793
prio     0.08620   1.09002  0.02887   0.02907  2.97   0.003  **
employed.percent -0.00770   0.99233  0.00311   0.00305 -2.52   0.012  *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
fin           0.684       1.462    0.466    1.004
age           0.950       1.053    0.903    1.000
race          1.362       0.734    0.765    2.424
wexp           0.940       1.064    0.609    1.449
mar           0.706       1.417    0.330    1.507
paro           0.949       1.054    0.640    1.405
prio           1.090       0.917    1.030    1.154
employed.percent 0.992       1.008    0.986    0.998

Concordance= 0.658 (se = 0.027 )
Rsquare= 0.002 (max possible= 0.066 )
Likelihood ratio test= 39.7 on 8 df, p=0.0000036
Wald test = 38.6 on 8 df, p=0.00000583
Score (logrank) test = 38.9 on 8 df, p=0.00000523, Robust = 33.7 p=0.0000458

(Note: the likelihood ratio and score tests assume independence of
      observations within a cluster, the Wald and robust score tests do not).
```

Professor Fox’s fold function

- <http://cran.r-project.org/doc/contrib/Fox-Companion/appendix-cox-regression.pdf>

Professor Fox’s fold function explained in the PDF above is handy in creating the dataset.


```
fold <- function(data, time, event, cov,
  cov.names=paste('covariate', '.', 1:ncovs, sep=""),
  suffix='.time', cov.times=0:ncov, common.times=TRUE, lag=0){
## Arguments:
## data: A data frame or numeric matrix (with column names) to be `unfolded.`
##       For reasons of efficiency, if there are factors in data these will be
##       converted to numeric variables in the output data frame.
## time: The quoted name of the event/censoring-time variable in data.
## event: The quoted name of the event/censoring-indicator variable in data.
## cov: A vector giving the column numbers of the time-dependent covariate
##       in data, or a list of vectors if there is more than one time-varying
##       covariate.
## cov.names: A character string or character vector giving the name or names
##             to be assigned to the time-dependent covariate(s) in the output data set.
## suffix: The suffix to be attached to the name of the time-to-event variable
##          in the output data set; defaults to '.time'.
## cov.times: The observation times for the covariate values, including the start
##             time. This argument can take several forms:
##             * The default is integers from 0 to the number of covariate values (i.e.,
##               one more than the length of each vector in cov).
##             * An arbitrary numerical vector with one more entry than the length of each
##               vector in cov.
##             * The columns in the input data set that give the observations times for each
##               individual. There should be one more column than the length of each
##               vector in cov.
##             * common.times: A logical value indicating whether the times of observation
##               are the same for all individuals; defaults to TRUE.
## lag: Number of observation periods to lag each value of the time-varying
##      covariate(s); defaults to 0.
vlag <- function(x, lag) c(rep(NA, lag), x[1:(length(x)-lag)])
xlag <- function(x, lag) apply(as.matrix(x), 2, vlag, lag=lag)
all.cov <- unlist(cov)
if (!is.list(cov)) cov <- list(cov)
ncovs <- length(cov)
nrow <- nrow(data)
ncol <- ncol(data)
ncov <- length(cov[[1]])
nobs <- nrow*ncov
if (length(unique(c(sapply(cov, length), length(cov.times)-1))) > 1)
  stop(paste(
    "all elements of cov must be of the same length and \n",
    "cov.times must have one more entry than each element of cov."))
var.names <- names(data)
subjects <- rownames(data)
omit.cols <- if (!common.times) c(all.cov, cov.times) else all.cov
keep.cols <- (1:ncol)[-omit.cols]
nkeep <- length(keep.cols)
if (is.numeric(event)) event <- var.names[event]
times <- if (common.times) matrix(cov.times, nrow, ncov+1, byrow=T)
  else data[, cov.times]
new.data <- matrix(Inf, nobs, 3 + ncovs + nkeep)
rownames <- rep("", nobs)
colnames(new.data) <- c('start', 'stop', paste(event, suffix, sep=""),
  var.names[-omit.cols], cov.names)
end.row <- 0
for (i in 1:nrow){
  start.row <- end.row + 1
  end.row <- end.row + ncov
  start <- times[i, 1:ncov]
  stop <- times[i, 2:(ncov+1)]
  event.time <- ifelse (stop == data[i, time] & data[i, event] == 1, 1, 0)
  keep <- matrix(unlist(data[i, -omit.cols]), ncov, nkeep, byrow=T)
  select <- apply(matrix(!is.na(data[i, all.cov])), ncol=ncovs), 1, all)
  rows <- start.row:end.row
  cov.mat <- xlag(matrix(unlist(data[i, all.cov]), nrow=length(rows)), lag)
  new.data[rows[select], ] <-
    cbind(start, stop, event.time, keep, cov.mat)[select,]
  rownames[rows] <- paste(subjects[i], '.', seq(along=rows), sep="")
}
row.names(new.data) <- rownames
as.data.frame(new.data[new.data[, 1] != Inf &
  apply(as.matrix(!is.na(new.data[, cov.names])), 1, all), ])
}
```

Use of lag is easy

```
Rossi.long2 <- fold(Rossi, time='week', event='arrest', cov=11:62, cov.names='emp', lag = 1)
Rossi.long2
```

	start	stop	arrest.time	week	arrest	fin	age	race	wexp	mar	paro	prio	educ	emp
1.2	1	2	0	20	1	0	27	1	0	0	1	3	3	0
1.3	2	3	0	20	1	0	27	1	0	0	1	3	3	0
1.4	3	4	0	20	1	0	27	1	0	0	1	3	3	0
1.5	4	5	0	20	1	0	27	1	0	0	1	3	3	0
1.6	5	6	0	20	1	0	27	1	0	0	1	3	3	0
1.7	6	7	0	20	1	0	27	1	0	0	1	3	3	0
1.8	7	8	0	20	1	0	27	1	0	0	1	3	3	0
1.9	8	9	0	20	1	0	27	1	0	0	1	3	3	0
1.10	9	10	0	20	1	0	27	1	0	0	1	3	3	0
1.11	10	11	0	20	1	0	27	1	0	0	1	3	3	0
1.12	11	12	0	20	1	0	27	1	0	0	1	3	3	0

1.13	12	13	0	20	1	0	27	1	0	0	1	3	3	0
1.14	13	14	0	20	1	0	27	1	0	0	1	3	3	0
1.15	14	15	0	20	1	0	27	1	0	0	1	3	3	0
1.16	15	16	0	20	1	0	27	1	0	0	1	3	3	0
1.17	16	17	0	20	1	0	27	1	0	0	1	3	3	0
1.18	17	18	0	20	1	0	27	1	0	0	1	3	3	0
1.19	18	19	0	20	1	0	27	1	0	0	1	3	3	0
1.20	19	20	1	20	1	0	27	1	0	0	1	3	3	0
2.2	1	2	0	17	1	0	18	1	0	0	1	8	4	0
2.3	2	3	0	17	1	0	18	1	0	0	1	8	4	0
2.4	3	4	0	17	1	0	18	1	0	0	1	8	4	0
2.5	4	5	0	17	1	0	18	1	0	0	1	8	4	0
2.6	5	6	0	17	1	0	18	1	0	0	1	8	4	0
2.7	6	7	0	17	1	0	18	1	0	0	1	8	4	0
2.8	7	8	0	17	1	0	18	1	0	0	1	8	4	0
2.9	8	9	0	17	1	0	18	1	0	0	1	8	4	0
2.10	9	10	0	17	1	0	18	1	0	0	1	8	4	0
2.11	10	11	0	17	1	0	18	1	0	0	1	8	4	1
2.12	11	12	0	17	1	0	18	1	0	0	1	8	4	1
2.13	12	13	0	17	1	0	18	1	0	0	1	8	4	1
2.14	13	14	0	17	1	0	18	1	0	0	1	8	4	1
2.15	14	15	0	17	1	0	18	1	0	0	1	8	4	1
2.16	15	16	0	17	1	0	18	1	0	0	1	8	4	0
2.17	16	17	1	17	1	0	18	1	0	0	1	8	4	0
3.2	1	2	0	25	1	0	19	0	1	0	1	13	3	0
3.3	2	3	0	25	1	0	19	0	1	0	1	13	3	0
3.4	3	4	0	25	1	0	19	0	1	0	1	13	3	0
3.5	4	5	0	25	1	0	19	0	1	0	1	13	3	0
3.6	5	6	0	25	1	0	19	0	1	0	1	13	3	0
3.7	6	7	0	25	1	0	19	0	1	0	1	13	3	0
3.8	7	8	0	25	1	0	19	0	1	0	1	13	3	0
3.9	8	9	0	25	1	0	19	0	1	0	1	13	3	0
3.10	9	10	0	25	1	0	19	0	1	0	1	13	3	0
3.11	10	11	0	25	1	0	19	0	1	0	1	13	3	0
3.12	11	12	0	25	1	0	19	0	1	0	1	13	3	0
3.13	12	13	0	25	1	0	19	0	1	0	1	13	3	0
3.14	13	14	0	25	1	0	19	0	1	0	1	13	3	0
3.15	14	15	0	25	1	0	19	0	1	0	1	13	3	0
3.16	15	16	0	25	1	0	19	0	1	0	1	13	3	0
3.17	16	17	0	25	1	0	19	0	1	0	1	13	3	0
3.18	17	18	0	25	1	0	19	0	1	0	1	13	3	1
3.19	18	19	0	25	1	0	19	0	1	0	1	13	3	0
3.20	19	20	0	25	1	0	19	0	1	0	1	13	3	0
3.21	20	21	0	25	1	0	19	0	1	0	1	13	3	0
3.22	21	22	0	25	1	0	19	0	1	0	1	13	3	0
3.23	22	23	0	25	1	0	19	0	1	0	1	13	3	0
3.24	23	24	0	25	1	0	19	0	1	0	1	13	3	0
3.25	24	25	1	25	1	0	19	0	1	0	1	13	3	0
4.2	1	2	0	52	0	1	23	1	1	1	1	1	5	0
4.3	2	3	0	52	0	1	23	1	1	1	1	1	5	0
4.4	3	4	0	52	0	1	23	1	1	1	1	1	5	0
4.5	4	5	0	52	0	1	23	1	1	1	1	1	5	0
4.6	5	6	0	52	0	1	23	1	1	1	1	1	5	1
4.7	6	7	0	52	0	1	23	1	1	1	1	1	5	1
4.8	7	8	0	52	0	1	23	1	1	1	1	1	5	1
4.9	8	9	0	52	0	1	23	1	1	1	1	1	5	1
4.10	9	10	0	52	0	1	23	1	1	1	1	1	5	1
4.11	10	11	0	52	0	1	23	1	1	1	1	1	5	1
4.12	11	12	0	52	0	1	23	1	1	1	1	1	5	1
4.13	12	13	0	52	0	1	23	1	1	1	1	1	5	1
4.14	13	14	0	52	0	1	23	1	1	1	1	1	5	1
4.15	14	15	0	52	0	1	23	1	1	1	1	1	5	1
4.16	15	16	0	52	0	1	23	1	1	1	1	1	5	1
4.17	16	17	0	52	0	1	23	1	1	1	1	1	5	1
4.18	17	18	0	52	0	1	23	1	1	1	1	1	5	1
4.19	18	19	0	52	0	1	23	1	1	1	1	1	5	1
4.20	19	20	0	52	0	1	23	1	1	1	1	1	5	1
4.21	20	21	0	52	0	1	23	1	1	1	1	1	5	1
4.22	21	22	0	52	0	1	23	1	1	1	1	1	5	1
4.23	22	23	0	52	0	1	23	1	1	1	1	1	5	0
4.24	23	24	0	52	0	1	23	1	1	1	1	1	5	0
4.25	24	25	0	52	0	1	23	1	1	1	1	1	5	0
4.26	25	26	0	52	0	1	23	1	1	1	1	1	5	0
4.27	26	27	0	52	0	1	23	1	1	1	1	1	5	0
4.28	27	28	0	52	0	1	23	1	1	1	1	1	5	0
4.29	28	29	0	52	0	1	23	1	1	1	1	1	5	0
4.30	29	30	0	52	0	1	23	1	1	1	1	1	5	0
4.31	30	31	0	52	0	1	23	1	1	1	1	1	5	0
4.32	31	32	0	52	0	1	23	1	1	1	1	1	5	0
4.33	32	33	0	52	0	1	23	1	1	1	1	1	5	1
4.34	33	34	0	52	0	1	23	1	1	1	1	1	5	1
4.35	34	35	0	52	0	1	23	1	1	1	1	1	5	1
4.36	35	36	0	52	0	1	23	1	1	1	1	1	5	1
4.37	36	37	0	52	0	1	23	1	1	1	1	1	5	1
4.38	37	38	0	52	0	1	23	1	1	1	1	1	5	1
4.39	38	39	0	52	0	1	23	1	1	1	1	1	5	1
4.40	39	40	0	52	0	1	23	1	1	1	1	1	5	1
4.41	40	41	0	52	0	1	23	1	1	1	1	1	5	1
4.42	41	42	0	52	0	1	23	1	1	1	1	1	5	1
4.43	42	43	0	52	0	1	23	1	1	1	1	1	5	1
4.44	43	44	0	52	0	1	23	1	1	1	1	1	5	1

4.45	44	45	0	52	0	1	23	1	1	1	1	1	5	1
4.46	45	46	0	52	0	1	23	1	1	1	1	1	5	1
4.47	46	47	0	52	0	1	23	1	1	1	1	1	5	1
4.48	47	48	0	52	0	1	23	1	1	1	1	1	5	1
4.49	48	49	0	52	0	1	23	1	1	1	1	1	5	1
4.50	49	50	0	52	0	1	23	1	1	1	1	1	5	1
4.51	50	51	0	52	0	1	23	1	1	1	1	1	5	1
4.52	51	52	0	52	0	1	23	1	1	1	1	1	5	1
5.2	1	2	0	52	0	0	19	0	1	0	1	3	3	0
5.3	2	3	0	52	0	0	19	0	1	0	1	3	3	0
5.4	3	4	0	52	0	0	19	0	1	0	1	3	3	0
5.5	4	5	0	52	0	0	19	0	1	0	1	3	3	0
5.6	5	6	0	52	0	0	19	0	1	0	1	3	3	0
5.7	6	7	0	52	0	0	19	0	1	0	1	3	3	0
5.8	7	8	0	52	0	0	19	0	1	0	1	3	3	0
5.9	8	9	0	52	0	0	19	0	1	0	1	3	3	0
5.10	9	10	0	52	0	0	19	0	1	0	1	3	3	0
5.11	10	11	0	52	0	0	19	0	1	0	1	3	3	1
5.12	11	12	0	52	0	0	19	0	1	0	1	3	3	1
5.13	12	13	0	52	0	0	19	0	1	0	1	3	3	1
5.14	13	14	0	52	0	0	19	0	1	0	1	3	3	1
5.15	14	15	0	52	0	0	19	0	1	0	1	3	3	1
5.16	15	16	0	52	0	0	19	0	1	0	1	3	3	1
5.17	16	17	0	52	0	0	19	0	1	0	1	3	3	1
5.18	17	18	0	52	0	0	19	0	1	0	1	3	3	1
5.19	18	19	0	52	0	0	19	0	1	0	1	3	3	1
5.20	19	20	0	52	0	0	19	0	1	0	1	3	3	1
5.21	20	21	0	52	0	0	19	0	1	0	1	3	3	1
5.22	21	22	0	52	0	0	19	0	1	0	1	3	3	1
5.23	22	23	0	52	0	0	19	0	1	0	1	3	3	1
5.24	23	24	0	52	0	0	19	0	1	0	1	3	3	1
5.25	24	25	0	52	0	0	19	0	1	0	1	3	3	1
5.26	25	26	0	52	0	0	19	0	1	0	1	3	3	1
5.27	26	27	0	52	0	0	19	0	1	0	1	3	3	1
5.28	27	28	0	52	0	0	19	0	1	0	1	3	3	1
5.29	28	29	0	52	0	0	19	0	1	0	1	3	3	1
5.30	29	30	0	52	0	0	19	0	1	0	1	3	3	1
5.31	30	31	0	52	0	0	19	0	1	0	1	3	3	1
5.32	31	32	0	52	0	0	19	0	1	0	1	3	3	1
5.33	32	33	0	52	0	0	19	0	1	0	1	3	3	1

[reached getOption("max.print") -- omitted 19235 rows]