Survival Regression Models

David M. Rocke

March, 2024

Background on the Proportional Hazards Model

The exponential distribution has constant hazard

$$\begin{aligned} f(t) &= \lambda e^{-\lambda t} \\ S(t) &= e^{-\lambda t} \\ h(t) &= \lambda \end{aligned}$$

Let's make two generalizations. First, let the hazard depend on covariates $x_1, x_2, \ldots x_p$. Second, let the base hazard depend on t but not on the covariates.

The Cox Model

The generalization is that the hazard function is

$$\begin{aligned} \eta &= \beta_1 x_1 + \cdots + \beta_p x_p \\ h(t|\text{covariates}) &= h_0(t) e^{\eta} \end{aligned}$$

This has a log link as in a generalized linear model. It is semi-parametric because the linear predictor depends on estimated parameters but the base hazard function is unspecified. There is no constant term because it is absorbed in the base hazard. Note that for two different individuals with possibly different covariates, the ratio of the hazard functions is $\exp(\eta_1)/\exp(\eta_2) = \exp(\eta_1 - \eta_2)$ which does not depend on t.

The coxph() Function in R

```
coxph {survival} R Documentation
Fit Proportional Hazards Regression Model

Description

Fits a Cox proportional hazards regression model.
Time dependent variables, time dependent strata, multiple events per subject,
and other extensions are incorporated using the counting process formulation of
Andersen and Gill.

Usage

coxph(formula, data=, weights, subset,
      na.action, init, control,
      ties=c("efron","breslow","exact"),
      singular.ok=TRUE, robust=FALSE,
      model=FALSE, x=FALSE, y=TRUE, tt, method, ...)
```

The coxph() Function in R

```
coxph(formula, data=, weights, subset,
      na.action, init, control,
      ties=c("efron","breslow","exact"),
      singular.ok=TRUE, robust=FALSE,
      model=FALSE, x=FALSE, y=TRUE, tt, method, ...)
Arguments

formula   a formula object, with the response on the left of a ~ operator,
          and the terms on the right. The response must be a survival object
          as returned by the Surv function.

data      a data.frame in which to interpret the variables named in the formula,
          or in the subset and the weights argument.

weights   vector of case weights. If weights is a vector of integers,
          then the estimated coefficients are equivalent to estimating the model
          from data with the individual cases replicated as many times as
          indicated by weights.

subset    expression indicating which subset of the rows of data
          should be used in the fit. All observations are included by default.
```

The coxph() Function in R

```
coxph(formula, data=, weights, subset,
      na.action, init, control,
      ties=c("efron","breslow","exact"),
      singular.ok=TRUE, robust=FALSE,
      model=FALSE, x=FALSE, y=TRUE, tt, method, ...)
Arguments

ties     a character string specifying the method for tie handling.
         If there are no tied death times all the methods are equivalent.
         Nearly all Cox regression programs use the Breslow method by default,
         but not this one. The Efron approximation is used as the default here,
         it is more accurate when dealing with tied death times, and is as
         efficient computationally. The "exact partial likelihood" is
         equivalent to a conditional logistic model, and is appropriate
         when the times are a small set of discrete values. See further below.
```

Cox Model for bmt Survival vs. Group

```
> dfsurv <- Surv(bmt$t2,bmt$d3)
> bmt.cox <- coxph(dfsurv~factor(group),data=bmt)
> summary(bmt.cox)

  n= 137, number of events= 83

                   coef exp(coef) se(coef)      z Pr(>|z|)
factor(group)2 -0.5742    0.5632   0.2873 -1.999   0.0457 *
factor(group)3  0.3834    1.4673   0.2674  1.434   0.1516
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

               exp(coef) exp(-coef) lower .95 upper .95
factor(group)2    0.5632     1.7757    0.3207     0.989
factor(group)3    1.4673     0.6815    0.8688     2.478

Concordance= 0.625  (se = 0.031 )
Rsquare= 0.094   (max possible= 0.996 )
Likelihood ratio test= 13.45  on 2 df,   p=0.001199
Wald test            = 13.03  on 2 df,   p=0.00148
Score (logrank) test = 13.81  on 2 df,   p=0.001004
```

Cox Model for bmt Survival vs. Group

Hypothesis tests for factor levels compare group 2 to group 1 and 3 to group 1. Group 3 has the highest hazard and group 2 has the lowest so the most significant comparison is not directly shown.

The coefficient 0.3834 is on the log-hazard-ratio scale, as in log-risk-ratio. The next column gives the hazard ratio 1.4673, and a hypothesis (Wald) test.

The (not shown) group 3 vs. group 2 log hazard ratio is 0.3834 + 0.5742 = 0.9576. The hazard ratio is then exp(0.9576) or 2.605. Inference on all coefficients and combinations can be constructed using coef(bmt.cox) and vcov(bmt.cox).

```
                  coef exp(coef) se(coef)      z Pr(>|z|)
factor(group)2 -0.5742    0.5632   0.2873 -1.999   0.0457 *
factor(group)3  0.3834    1.4673   0.2674  1.434   0.1516
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Cox Model for bmt Survival vs. Group

```
The next part of the output gives 95% confidence intervals for the
relative risk. For the difference between groups 2 and 3 we need to use
coef(bmt.cox) and vcov(bmt.cox).

                exp(coef) exp(-coef) lower .95 upper .95
factor(group)2     0.5632     1.7757    0.3207     0.989
factor(group)3     1.4673     0.6815    0.8688     2.478

> coef(bmt.cox)
factor(group)2 factor(group)3
    -0.5741967      0.3834137

> vcov(bmt.cox)
                factor(group)2 factor(group)3
factor(group)2     0.08254038     0.04181177
factor(group)3     0.04181177     0.07148991
```

Cox Model for bmt Survival vs. Group

```
                   coef exp(coef) se(coef)      z Pr(>|z|)
factor(group)2 -0.5742    0.5632   0.2873 -1.999   0.0457 *
factor(group)3  0.3834    1.4673   0.2674  1.434   0.1516
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

> coef(bmt.cox)
factor(group)2 factor(group)3
    -0.5741967      0.3834137
> vcov(bmt.cox)
               factor(group)2 factor(group)3
factor(group)2     0.08254038     0.04181177
factor(group)3     0.04181177     0.07148991

> sqrt(vcov(bmt.cox)[1,1])
[1] 0.2872984
> sqrt(vcov(bmt.cox)[2,2])
[1] 0.267376
```

Cox Model for bmt Survival vs. Group

```
                   coef exp(coef) se(coef)      z Pr(>|z|)
factor(group)2 -0.5742    0.5632   0.2873 -1.999   0.0457 *
factor(group)3  0.3834    1.4673   0.2674  1.434   0.1516

               exp(coef) exp(-coef) lower .95 upper .95
factor(group)2    0.5632     1.7757    0.3207     0.989
factor(group)3    1.4673     0.6815    0.8688     2.478

> c1 <- coef(bmt.cox)
> v1 <- vcov(bmt.cox)
> cont1 <- c(-1,1)
> t(cont1) %*% c1
          [,1]
[1,] 0.9576104                          log hazard ratio group 3 to group2
> t(cont1) %*% v1 %*% cont1
           [,1]
[1,] 0.07040675
> sqrt(t(cont1) %*% v1 %*% cont1)
          [,1]
[1,] 0.2653427                          standard error of log hazard ratio
```

Cox Model for bmt Survival vs. Group

```
                  coef exp(coef) se(coef)      z Pr(>|z|)
factor(group)2 -0.5742    0.5632   0.2873 -1.999   0.0457 *
factor(group)3  0.3834    1.4673   0.2674  1.434   0.1516

> t(cont1) %*% c1
[1,] 0.9576104                                  log hazard ratio group 3 to group2
> sqrt(t(cont1) %*% v1 %*% cont1)
[1,] 0.2653427                                  standard error of log hazard ratio
> lhr23 <- t(cont1) %*% c1
> sqrt(t(cont1) %*% v1 %*% cont1)
[1,] 0.2653427
> selhr23 <- sqrt(t(cont1) %*% v1 %*% cont1)
> lhr23/selhr23
[1,] 3.608957
> lhr23+1.96*selhr23
[1,] 1.477682
> lhr23-1.96*selhr23
[1,] 0.4375387
```

Cox Model for bmt Survival vs. Group

```
Concordance is agreement of first failure between pairs of subjects and
higher predicted risk between those subjects, omitting non-informative pairs.

The Rsquare value is Cox and Snell's pseudo R-squared and is not very useful.

There are three tests for whether the model with the group covariate
is better than the one without
--Usual likelihood ratio chi-squared
--Wald test chi-squared, obtained with the covariance matrix
--score = log-rank test, as with comparison of survival functions.
The likelihood ratio test is probably best in smaller samples, followed
by the Wald test.

Concordance= 0.625  (se = 0.031 )
Rsquare= 0.094   (max possible= 0.996 )
Likelihood ratio test= 13.45  on 2 df,   p=0.001199
Wald test            = 13.03  on 2 df,   p=0.00148
Score (logrank) test = 13.81  on 2 df,   p=0.001004
```

Cox Model for bmt Survival vs. Group

```
> c1 <- coef(bmt.cox)
> v1 <- vcov(bmt.cox)
> t(c1)%*%solve(v1)%*%c1
         [,1]
[1,] 13.03152
```

```
Concordance= 0.625  (se = 0.031 )
Rsquare= 0.094   (max possible= 0.996 )
Likelihood ratio test= 13.45  on 2 df,   p=0.001199
Wald test            = 13.03  on 2 df,   p=0.00148
Score (logrank) test = 13.81  on 2 df,   p=0.001004
```

A general Wald test for $H_0 : \beta = \beta_0$ is obtained with

$$(\hat{\beta} - \beta_0)^\top V^{-1} (\hat{\beta} - \beta_0)$$

Cox Model for bmt Survival vs. Group

```
The anova function performs the likelihood ratio test for comparing models.
One can use drop1(), add1(), step(), or compare two explicit models.

> anova(bmt.cox)
Analysis of Deviance Table
 Cox model: response is dfsurv
Terms added sequentially (first to last)

               loglik  Chisq Df Pr(>|Chi|)
NULL          -373.30
factor(group) -366.57 13.452  2   0.001199 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



Concordance= 0.625  (se = 0.031 )
Rsquare= 0.094   (max possible= 0.996 )
Likelihood ratio test= 13.45  on 2 df,   p=0.001199
Wald test            = 13.03  on 2 df,   p=0.00148
Score (logrank) test = 13.81  on 2 df,   p=0.001004
```

Inference on Combinations of Coefficients

If a predictor is categorical, with possible values $a_1, a_2, \ldots a_p$ then the output of essentially any regression method is in terms of $p-1$ coefficients $B = (b_2, b_3, \ldots, b_p)$, which are differences on the appropriate scale of groups $2, 3, \ldots, p$ from group 1. and an estimated covariance matrix V. Any linear combination of coefficients, such as $b_3 - b_2$ can be represented via a vector of weights w of length $p-1$ such as $(-1, 1, 0, \ldots, 0)$ in the form $w^\top B$ whose variance is then $w^\top V w$.

Survival Curves from the Cox Model

We defined the "base hazard" but then did not use it to estimate the coefficients. In fact, there is no meaningful *base* hazard. In this case, it is the hazard for group 1. We can use `survfit` to get survival functions, but by default it produces the hazard for an average individual with average covariates. This is like the family with 1.5 children—it does not exist. Always it is best to specify the covariate level(s) for which you want the survival curve(s). In this case, we can plot the Cox model survival curves, which are by definition proportional, along with the individual KM curves for the groups.

```
plot(survfit(dfsurv~group,data=bmt))
lines(survfit(bmt.cox,data.frame(group=1:3),conf.int=F),col="red")
legend("bottomleft",c("Kaplan-Meier","Cox Model"),col=c("black","red"),lwd=1)
title("Survival Functions for Three Groups by KM and Cox Model")
```

When we use survfit() with a Cox model, we should include a data frame with the same named columns as the predictors in the Cox model and one or more levels of each.

For example

```
> data.frame(group=1:3,age=50)
  group age
1     1  50
2     2  50
3     3  50
```

```
> data.frame(group=rep(1:3,2),age=rep(c(50,60),each=3))
  group age
1     1  50
2     2  50
3     3  50
4     1  60
5     2  60
6     3  60
```

Maximum Partial Likelihood

If subject $j(i)$ is the one who fails at time t_i, then the *partial likelihood* is

$$L(\beta|T) = \prod_i \frac{\theta_{j(i)}}{\sum_{k \in R(t_i)} \theta_k}$$

where T stands for all the data including times, censoring, and covariate values, while β is the vector of coefficients.

The *partial log likelihood* is

$$\ell\ell(\beta|T) = \sum_i \left[\ln[\theta_{j(i)}] - \ln \left(\sum_{k \in R(t_i)} \theta_k \right) \right]$$

and with $\theta_k = \exp(\eta_k)$, let

$$\begin{aligned} \theta_k^m &= \frac{\partial}{\partial \beta_m} \theta_k \\ &= \theta_k \frac{\partial}{\partial \beta_m} \eta_k \\ &= \theta_k x_{mk} \end{aligned}$$

Then

$$\ell\ell(\beta|T) = \sum_i \left[\ln[\theta_{j(i)}] - \ln\left(\sum_{k \in R(t_i)} \theta_k \right) \right]$$

$$\frac{\partial}{\partial \beta_m} \ell\ell(\beta|T) = \sum_i [\theta_{j(i)}]^{-1} \theta^m_{j(i)} - \left[\sum_{k \in R(t_i)} \theta_k \right]^{-1} \sum_{k \in R(t_i)} \theta^m_k$$

which is the gradient vector, AKA the score statistic, and similarly we can derive the Hessian, whose negative inverse is the Fisher information. This can be used with a variety of optimization techniques such as Newton's method to find the MLE. Similar calculations can be used with tied event times.

In logistic regression, if all the values of a covariate for cases are larger than all the values for controls, or the reverse, then the covariate is very predictive, but the coefficient will diverge in estimation to $\pm\infty$. The same happens in Cox regression if the covariate values for the individuals with events are increasing (or decreasing) as the event times go from smallest to largest.
Paradoxically, this makes the numerical optimization invalid while strongly indicating the covariate is related to risk.

Wald Tests

We use the maximum partial likelihood estimates $\hat{\beta}$ of the parameter vector β which has estimated covariance matrix V from the Fisher information. The diagonal entries of V are the squares of the standard errors which we can use for tests and confidence intervals for single coefficients (these are given in the output). A linear combination $c^\top \hat{\beta}$ of coefficients has covariance matrix $c^\top V c$. The hypothesis $H_0 : \hat{\beta} = \beta_0$ can be tested with $(\hat{\beta} - \beta_0)^\top V (\hat{\beta} - \beta_0)$ which is asymptotically $\chi^2(p)$ under the null, where p is the number of parameters.

Likelihood Ratio Tests

Asymptotically, the log likelihood ratio $2[\ell\ell(\hat{\beta}) - \ell\ell(\beta_0)]$ is $\chi^2(p)$. This test, as well as the Wald test, can be used with partial specification by the null hypothesis of the coefficients in which case the dimension of the χ^2 statistic is the number of linear constraints. For example, if one coefficient is specified to be zero, this is equivalent to leaving that variable out and re-running the optimization. That is a test of that one coefficient and has dimension 1. Note that the Wald test of one coefficient uses the previous coefficient estimates, whereas the LR test re-estimates all the coefficients.

Score Tests

The score statistic AKA the gradient is 0 at the MLE. Let $G(\beta)$ be the score statistic, so that $G(\hat{\beta}) = 0$. The statistic

$$G(\beta_0)^\top V G(\beta_0)$$

has asymptotically a $\chi^2(p)$ distribution as well. If there are no ties, then the score test and the log rank test (as used in `survdiff`) are the same. In many cases, the LR test has faster convergence than the Wald test, though the book indicates that they are similar. The score test is generally less accurate.

Coding and Transforming Predictors

- A factor is a categorical covariate. If it has two levels, and those are coded as 0 and 1 in a numerical variable, then the coefficient is the predicted difference in the two levels (such as male/female).
- If there are more than two levels, then one can represent the factor with one fewer predictors than the number of levels. For example if the coding is group $= 1, 2, 3$, then we could define $x_1 = 1$ iff group $=2$ and $x_2 = 1$ iff group $= 3$.

Coding and Transforming Predictors

- This is rather old fashioned, though may sometimes be useful. Instead

  `dtype = factor(dtype,labels = c("NHL","HOD"))`

  redefines the variable to be a *factor*, which is inherently categorical.
- The coefficients though are by default comparisons with the first level.

Coding and Transforming Predictors

- Numerical variables may be transformed linearly so that the coefficient is in interpretable units.
- It also may improve the model to use the log, square root, or inverse of the original variable.
- The urge to categorize numeric variables should be resisted unless there is strong evidence that it helps.
- Hemoglobin A1C test (from CDC web page):

$$\begin{cases} \text{Normal} & \text{A1C} < 5.7\% \\ \text{Prediabetes} & 5.7\% \leq \text{A1C} \leq 6.4\% \\ \text{Diabetes} & \text{A1C} \geq 6.5\% \end{cases}$$

Use of Tests in Model Building

Generally in a survival analysis we have chosen a response, such as progression-free survival, and perhaps a main predictor, such as drug vs. standard of care. We may have other covariates of interest which are thought possibly to influence survival, or even perhaps the efficacy of the drug (the latter would imply an interaction term). If a covariate is not statistically significant, does that mean we should remove it from the model? Well, not necessarily.

We could compare the models with a measure of predictive performance such as the Akaiki Information Criterion (AIC) or the Bayesian Information Criterion (BIC). We might keep a predictor in the model because it is useful or because other studies have used it.

For clinical trials, the analysis must be prespecified and usually consists of a simple comparison of treatment and standard of care (remembering that these trials usually are randomized). Secondary analysis, also usually prespecified, can encompass covariates.

ANOVA

The analysis of variance is in linear regression the division of the sums of squares into parts assigned to covariates and interactions as well as the total and the error term. More generally, this describes a comparison of two models in which one is derived from the other by omitting terms. This can be done for many types of regression models including the ones in `coxph`.

OED: (post-classical Latin) *analysis*: act of resolving (something) into its elements (13th cent. in British and continental sources).

KMsurv hodg data set

Data on lymphoma: Hodgins disease and non-Hodkins lymphoma:

```
gtype =   Graft type (1=allogenic, 2=autologous)
dtype =   Disease type (1=Non Hodgkin lymphoma, 2=Hodgkins disease)
time =    Time to death or relapse, days
delta =   Death/relapse indicator (0=alive, 1=dead)
score =   Karnofsky score
wtime =   Waiting time to transplant in months
```

Karnovsky score indicates general functionality of the individual. There are 43 patients, 20 with Hodgkin's disease and and 23 with non-Hodkins lymphoma.

```
> hodg.cox1 <- coxph(hodg.surv ~ gtype * dtype + score + wtime, data = hodg2)
> anova(hodg.cox1)
Analysis of Deviance Table
 Cox model: response is hodg.surv
Terms added sequentially (first to last)

             loglik   Chisq Df Pr(>|Chi|)
NULL        -87.258
gtype       -87.194  0.1285  1    0.71996
dtype       -86.995  0.3973  1    0.52848
score       -74.445 25.1003  1  5.442e-07 ***
wtime       -73.899  1.0920  1    0.29604
gtype:dtype -71.181  5.4357  1    0.01973 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

This sequential addition of variables is not the most useful. The test indicates the usefulness of the given variable in a model already including variables that proceed it in the list.

```
> drop1(hodg.cox1)
Single term deletions

Model:
hodg.surv ~ gtype * dtype + score + wtime
            Df    AIC
<none>         152.36                    Any drop increases the AIC (bad)
score        1 167.60
wtime        1 153.64                    Can't drop gtype or dtype
gtype:dtype  1 155.80                      if gtype:dtype is in the model

> drop1(hodg.cox1,test="Chisq")
Single term deletions

Model:
hodg.surv ~ gtype * dtype + score + wtime
            Df    AIC     LRT Pr(>Chi)
<none>         152.36
score        1 167.60 17.2365  3.3e-05 ***        score and gtype:dtype
wtime        1 153.64  3.2792  0.07016 .            are significant by LR test
gtype:dtype  1 155.80  5.4357  0.01973 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The command `drop1` respects hierarchy, meaning that if an interaction is in the model then none of the subsidiary terms can be dropped. A LR test can be added but the AIC is always given.

$$\text{AIC} = -2\ell\ell + 2p$$

where p is the effective number of parameters. When terms are added to a model, the $\ell\ell$ cannot drop, but when penalized by the dimension of the predictors it may.

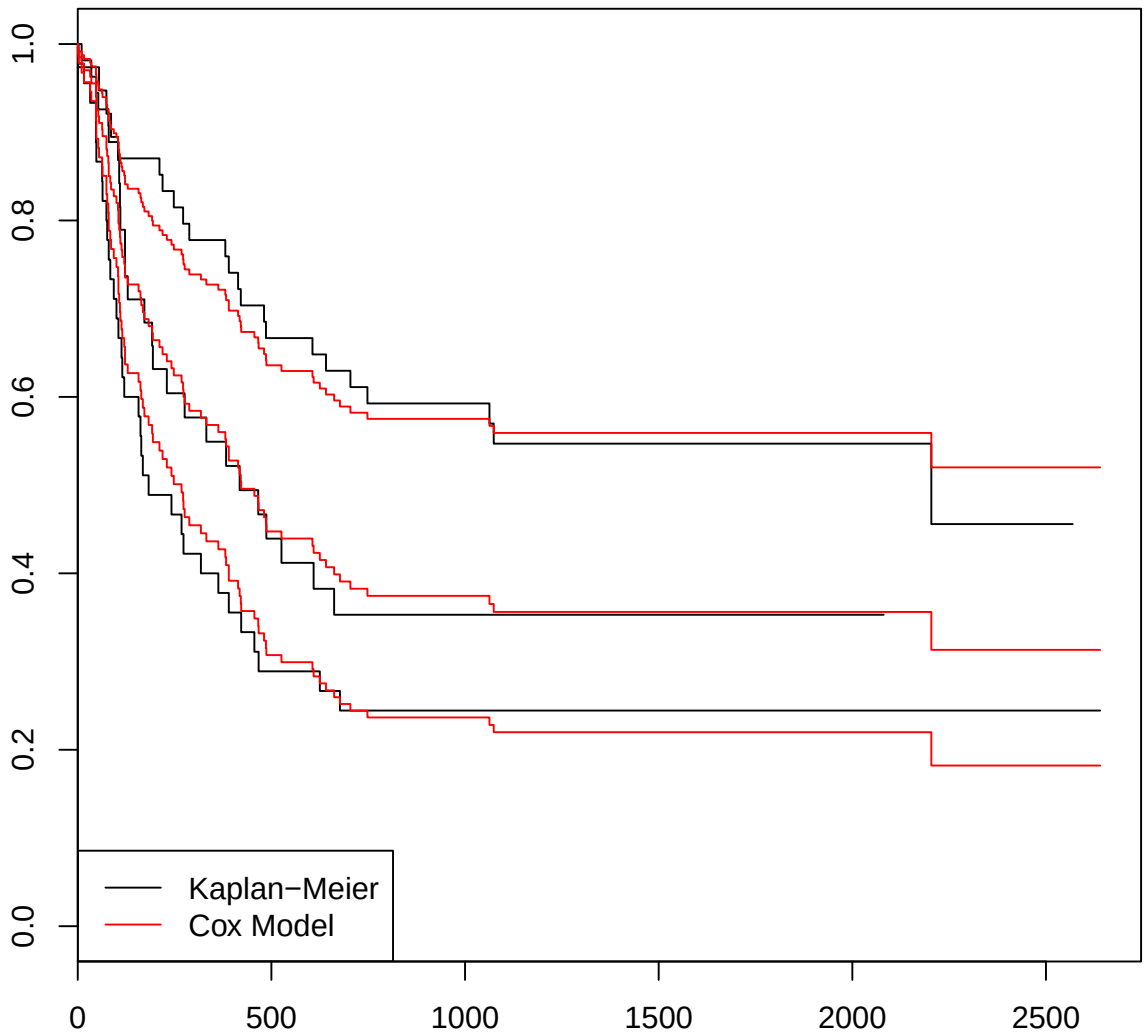
The penalty term is $2p$ by default but may be set to be kp. If $k = \ln(n)$ this is called the Bayesian Information Criterion (BIC) which favors smaller models than the AIC. Significance testing usually results in smaller models than the AIC or else the same model.

If there are missing values, then the models may be fit to different data sets which makes the inference invalid.

There is also an `add1` command which requires a term indicating the largest possible model that can be considered. An example is on the next slide.

```
> add1(hodg.cox1,scope = ~ gtype * dtype * score * wtime,test="Chisq")
Single term additions

Model:
hodg.surv ~ gtype * dtype + score + wtime
            Df    AIC    LRT Pr(>Chi)
<none>         152.36
gtype:score  1 149.03 5.3339  0.02091 *
dtype:score  1 153.13 1.2355  0.26634
gtype:wtime  1 153.13 1.2321  0.26699
dtype:wtime  1 154.27 0.0884  0.76619
score:wtime  1 154.30 0.0630  0.80188
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```


Survival Functions for Three Groups by KM and Cox Model
1.0
0.8
0.6
0.4
0.2
0.0
0
500
1000
1500
2000
2500
Kaplan−Meier
Cox Model