Survival analysis

Andrew Roth

2024-06-19

Motivation

One of the more common analyses in biomedical applications is survival or time to failure analysis. Such analyses are useful when considering how a perturbation impacts the time it takes for an event to occur. For example, we could ask whether giving mice a drug increases the time to develop tumours. There are two key variables in all survival analysis data.

- time - The time that a subject has been observed until. This can be time until the event occurred, or if the event doesn't occur time that a subject was followed before the study ended.
- censor - Whether the observation has been censored. This essentially indicates whether an even has happened or not for a given subject. Typically the data will right censored because a study ends and for example some patients are alive. In this case we would indicate they are censored.

Additional variables relevant to the study will also be included. The goal of survival analysis is then to determine if these variables are associated with survival.

Univariate survival (Kaplan-Meier curves)

First we will consider only a single study variable. The most common way to analyse such data is to use Kaplan-Meier curves.

First, we will load some libraries for survival analysis and plotting.

```
library(ggplot2)
library(survival)
```

We will use the veteran dataset provided by the `survival` package.

```
df <- veteran
summary(df)
```

```
##       trt             celltype       time           status
##  Min.   :1.000   squamous :35   Min.   :  1.0   Min.   :0.0000
##  1st Qu.:1.000   smallcell:48   1st Qu.: 25.0   1st Qu.:1.0000
##  Median :1.000   adeno    :27   Median : 80.0   Median :1.0000
##  Mean   :1.496   large    :27   Mean   :121.6   Mean   :0.9343
##  3rd Qu.:2.000                  3rd Qu.:144.0   3rd Qu.:1.0000
##  Max.   :2.000                  Max.   :999.0   Max.   :1.0000
##      karno          diagtime           age            prior
##  Min.   :10.00   Min.   : 1.000   Min.   :34.00   Min.   : 0.00
##  1st Qu.:40.00   1st Qu.: 3.000   1st Qu.:51.00   1st Qu.: 0.00
##  Median :60.00   Median : 5.000   Median :62.00   Median : 0.00
##  Mean   :58.57   Mean   : 8.774   Mean   :58.31   Mean   : 2.92
##  3rd Qu.:75.00   3rd Qu.:11.000   3rd Qu.:66.00   3rd Qu.:10.00
##  Max.   :99.00   Max.   :87.000   Max.   :81.00   Max.   :10.00
```

Here status means censoring status or whether an event has occurred. The encoding is 0 means no event (censored) and 1 means an event (not censored). We can look at the number of such entries using the `table` command.

```r
table(df$status)
```

```
##
##   0   1
##   9 128
```

We see there are only 9 censored (0) values in this dataset.

Now we will fit the Kaplan-Meier curve.

```r
km_fit <- survfit(Surv(time, status) ~ 1, data=df)
```

To do the actual fitting we create a `Surv` object and pass it to the `survfit` function. The `survfit` function the returns the fitted KM curve object. The `Surv` function requires the column names for time and censoring, in this case `time` and `status`. For the `survfit` we pass in a formula where the `Surv` objects serves as the y and in this case nothing is used as x indicated by `~`. Said another way, we are building the overall survival curve independent of any variables in the dataset.

Next we can look at a summary of the survival curve.

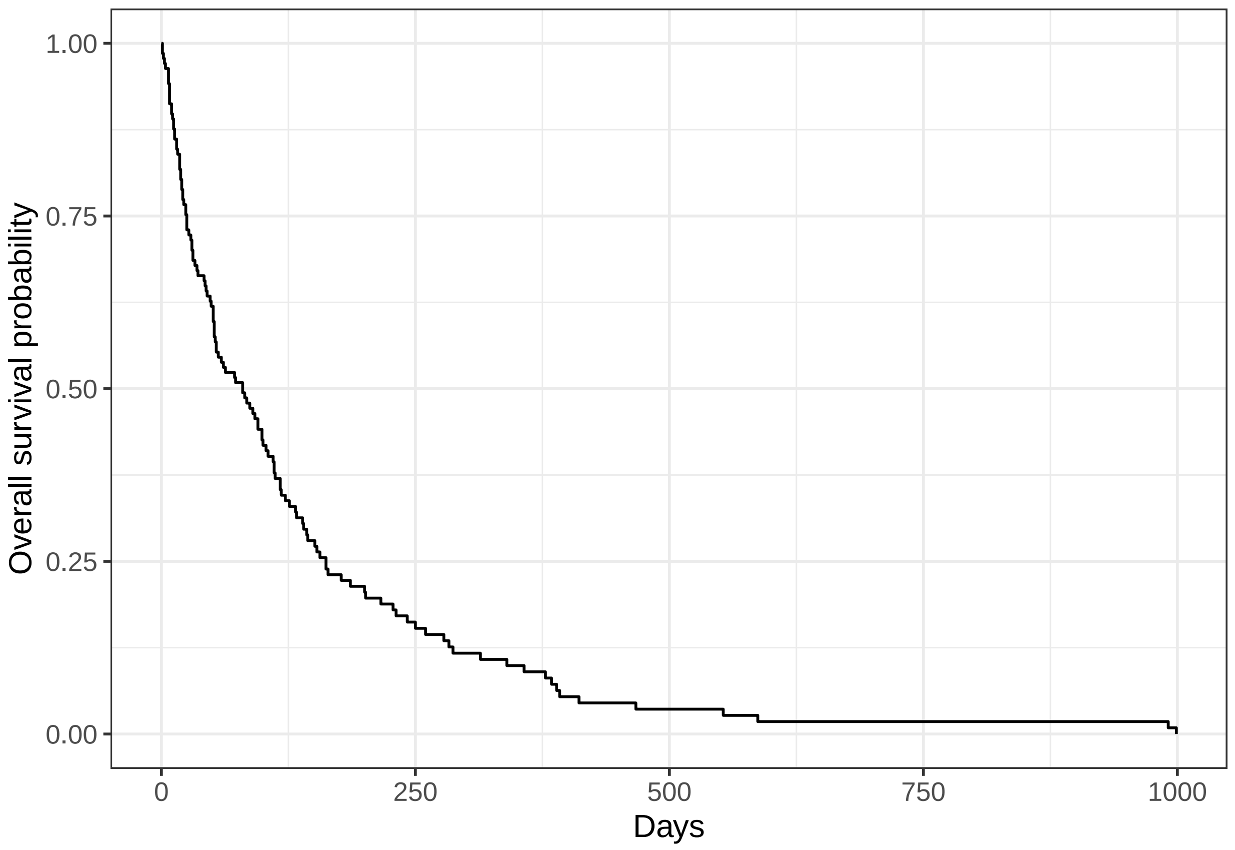
```r
summary(km_fit, times = c(1, 30, 60, 90*(1:10)))
```

```
## Call: survfit(formula = Surv(time, status) ~ 1, data = df)
##
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##     1    137       2    0.985  0.0102      0.96552       1.0000
##    30     97      39    0.700  0.0392      0.62774       0.7816
##    60     73      22    0.538  0.0427      0.46070       0.6288
##    90     62      10    0.464  0.0428      0.38731       0.5560
##   180     27      30    0.222  0.0369      0.16066       0.3079
##   270     16       9    0.144  0.0319      0.09338       0.2223
##   360     10       6    0.090  0.0265      0.05061       0.1602
##   450      5       5    0.045  0.0194      0.01931       0.1049
##   540      4       1    0.036  0.0175      0.01389       0.0934
##   630      2       2    0.018  0.0126      0.00459       0.0707
##   720      2       0    0.018  0.0126      0.00459       0.0707
##   810      2       0    0.018  0.0126      0.00459       0.0707
##   900      2       0    0.018  0.0126      0.00459       0.0707
```

Here we have passed the fitted curve, and a vector of times we wish to extract information about the curve from.

Now we can plot the survival curve using the `ggsurvfit` package which adds survival function support to ggplot.

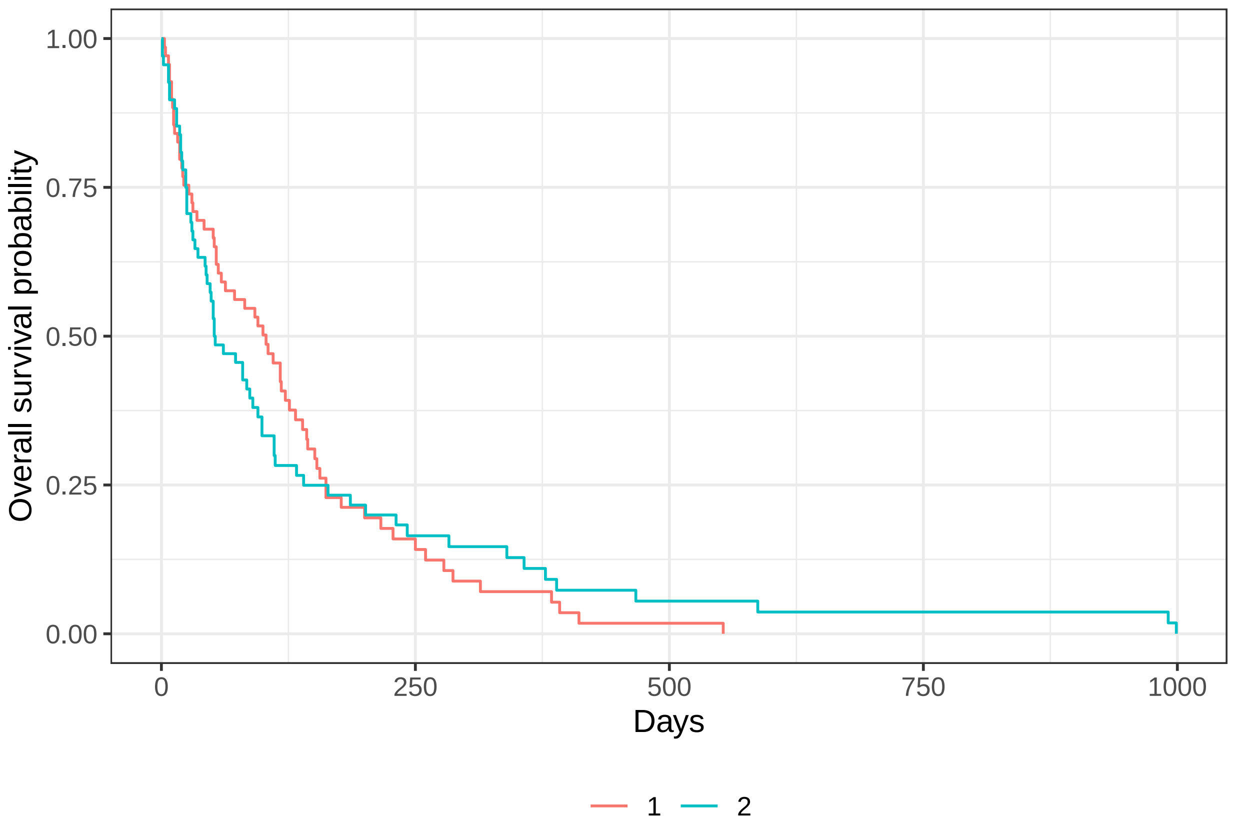
```r
library(ggsurvfit)

ggsurvfit(km_fit) +
  labs(
    x = "Days",
    y = "Overall survival probability"
  ) +
  ylim(0, 1)
```



So far we have only considered the overall survival, but typically we would want to consider how some variable impacts survival. Let's take a look at the treatment column `trt`, which has two values 1-standard and 2-test.

Like before we build the survival curve object, but now we alter the formula to indicate we want to vary by treatment. We will also do a quick cleanup and make `trt` a factor before doing this. One other minor thing, is to get plotting to play nice with our outputs we use the `survfit2` function provided by `ggsurvfit` now.

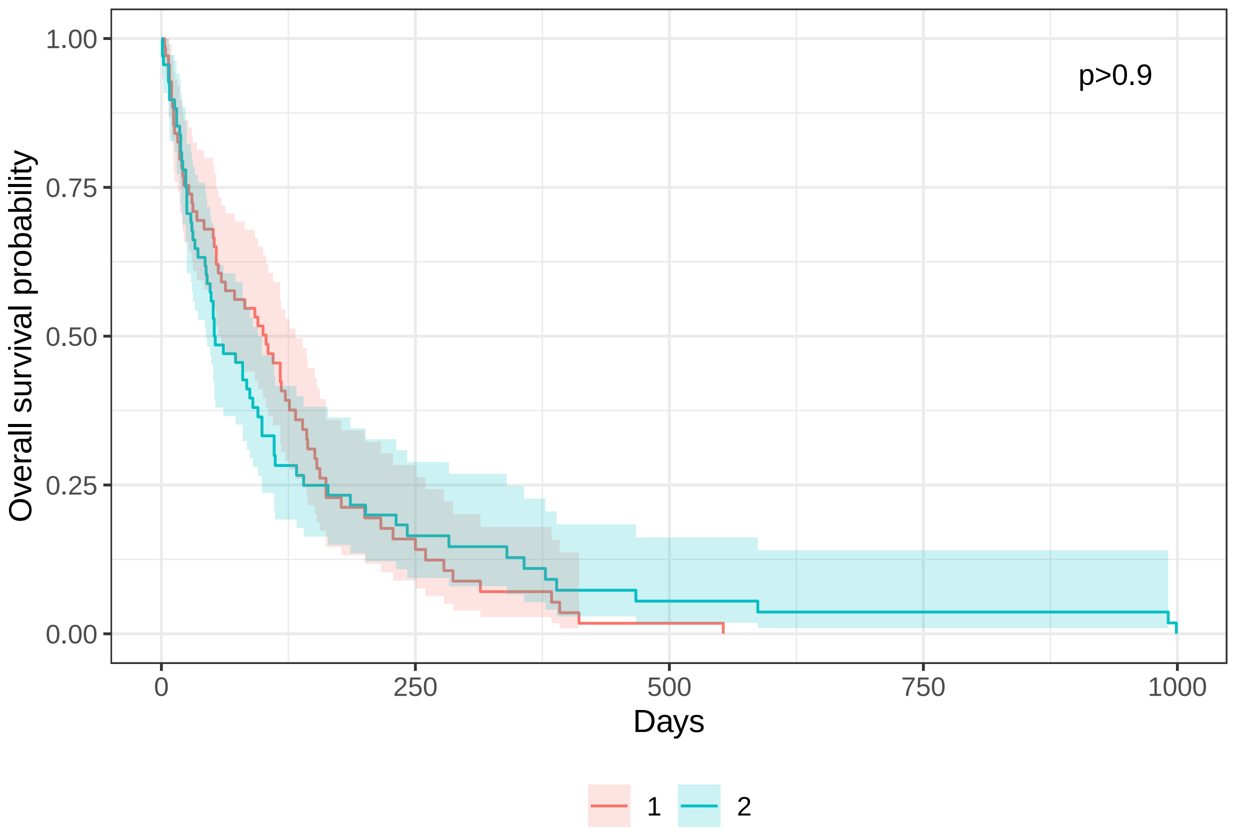
```r
df$trt <- factor(df$trt, levels=c(1, 2))

km_fit <- survfit2(Surv(time, status) ~ trt, data=df)

ggsurvfit(km_fit) +
  labs(
    x = "Days",
    y = "Overall survival probability"
  ) +
  ylim(0, 1)
```



Now we see separate curves for each treatment group.

We can also add p-values and confidence intervals to the plot.

```
ggsurvfit(km_fit) +
  add_pvalue(location="annotation") +
  add_confidence_interval() +
  labs(
    x = "Days",
    y = "Overall survival probability"
  ) +
  ylim(0, 1)
```



Multivariable survival

Often we want to consider whether a variable of interest is independently prognostic from other variables. In this case we can use the Cox proportional hazard model. Roughly, the Cox model is the survival equivalent of doing multivariable logistic regression.

Let's consider treatment, age and cell type our analysis. The code is similar to before, but now we use the `coxph` function for curve fitting.

```
df$celltype <- factor(df$celltype)

cox <- coxph(
  Surv(time, status) ~ trt + age + celltype,
  data=df
  )
summary(cox)
```

```
## Call:
## coxph(formula = Surv(time, status) ~ trt + age + celltype, data = df)
##
##   n= 137, number of events= 128
##
##                        coef exp(coef) se(coef)     z Pr(>|z|)
## trt2               0.179011  1.196034 0.201404 0.889    0.374
## age                0.004097  1.004106 0.009581 0.428    0.669
## celltypesmallcell 1.080310  2.945592 0.274647 3.933 8.37e-05 ***
## celltypeadeno      1.170470  3.223506 0.294727 3.971 7.15e-05 ***
## celltypelarge      0.292624  1.339939 0.285504 1.025    0.305
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                    exp(coef) exp(-coef) lower .95 upper .95
## trt2                   1.196     0.8361    0.8059     1.775
## age                    1.004     0.9959    0.9854     1.023
## celltypesmallcell      2.946     0.3395    1.7195     5.046
## celltypeadeno          3.224     0.3102    1.8091     5.744
## celltypelarge          1.340     0.7463    0.7657     2.345
##
## Concordance= 0.619  (se = 0.028 )
## Likelihood ratio test= 26.04  on 5 df,   p=9e-05
## Wald test            = 25.01  on 5 df,   p=1e-04
## Score (logrank) test = 26.51  on 5 df,   p=7e-05
```

We can generate a forest plot of the hazard ratios for the coefficients using the `survminer` library.

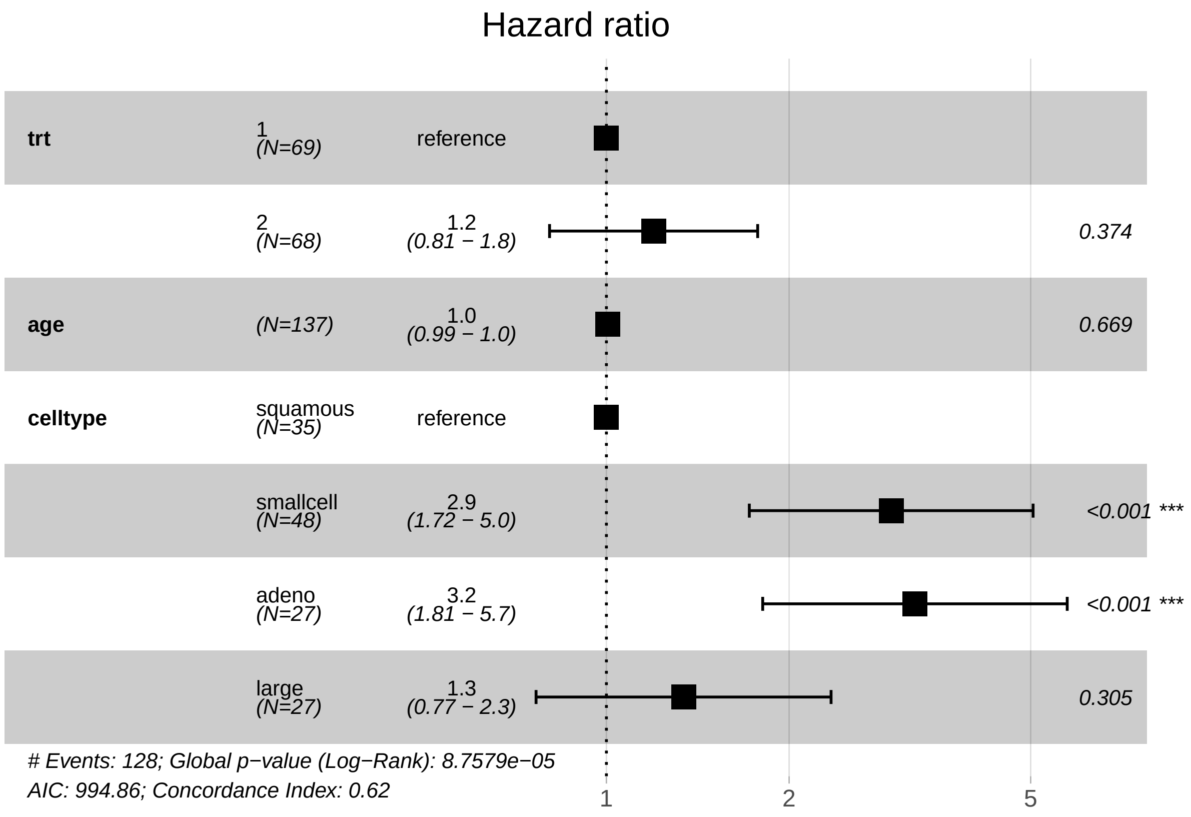
```
library(survminer)
```

```
## Loading required package: ggpubr
```

```
##
## Attaching package: 'survminer'
```

```
## The following object is masked from 'package:survival':
##
##     myeloma
```

```
ggforest(cox, data=df)
```



From this the treatment does appear to have an effect. However, cell type of the cancer does. In particular, smallcell and adeno cell types impose a higher risk versus the reference class of squamous.