

# Examples of focused model comparison: parametric survival models

Christopher Jackson

[chris.jackson@mrc-bsu.cam.ac.uk](mailto:chris.jackson@mrc-bsu.cam.ac.uk)

---

## Abstract

This vignette illustrates focused model comparison with the **fic** package for parametric survival models fitted with the **flexsurv** and **survival** packages. A challenge of this example is that the same models can be parameterised in multiple ways. For focused model comparison, the parameters need to be defined consistently between the models being compared. This might require a different parameterisation to be used when fitting a model or defining the focus function.

*Keywords:* models, survival.

---

## 1. Parametric survival models: example

Common model choice problems in parametric survival analysis include:

1. the selection of covariates, for example in a proportional hazards or accelerated failure time regression model.
2. the selection of the appropriate level of flexibility for a parametric hazard or survival function (given specific “baseline” covariate values).

In this simplified example, focused model comparison is used for the second of these problems. Using the **flexsurv** package ([Jackson 2016](#)), parametric distributions of increasing complexity are fitted to a set of 686 right-censored survival times from patients with primary node positive breast cancer (originally from [Sauerbrei and Royston \(1999\)](#), and also provided in **flexsurv**). The exponential, Weibull and generalized gamma models are fitted, which have one, two and three parameters respectively.

```
if (!require("flexsurv"))
  stop("The `flexsurv` package should be installed
to run code in this vignette")

## Loading required package: flexsurv
## Warning: package 'flexsurv' was built under R version 3.5.0
##
## Attaching package: 'flexsurv'
```

```
## The following object is masked from 'package:msm':
##
##      qgeneric

ex <- flexsurvreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="exponential")

## Warning in seq.default(along = temp): partial argument match of 'along' to
## 'along.with'
## Warning: partial match of 'coef' to 'coefficients'

we <- flexsurvreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="weibull")

## Warning in seq.default(along = temp): partial argument match of 'along' to
## 'along.with'
## Warning in seq.default(along = temp): partial match of 'coef' to 'coefficients'

gg <- flexsurvreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="gengamma")
```

The fitted survival curve from the generalised gamma model appears to match the Kaplan-Meier estimate most closely throughout the 7 years of follow up.

```
plot(gg, ci=FALSE, conf.int=FALSE, ylab="Survival", xlab="Years")

## Warning in seq.default(along = x$dlist$pars): partial argument match of 'along'
## to 'along.with'
## Warning in rep(col, length = nrow(X)): partial argument match of 'length' to
## 'length.out'
## Warning in rep(lty, length = nrow(X)): partial argument match of 'length' to
## 'length.out'
## Warning in rep(lwd, length = nrow(X)): partial argument match of 'length' to
## 'length.out'
## Warning in rep(col.ci, length = nrow(X)): partial argument match of 'length'
## to 'length.out'
## Warning in rep(lty.ci, length = nrow(X)): partial argument match of 'length'
## to 'length.out'
## Warning in rep(lwd.ci, length = nrow(X)): partial argument match of 'length'
## to 'length.out'

lines(we, col="blue", ci=FALSE)

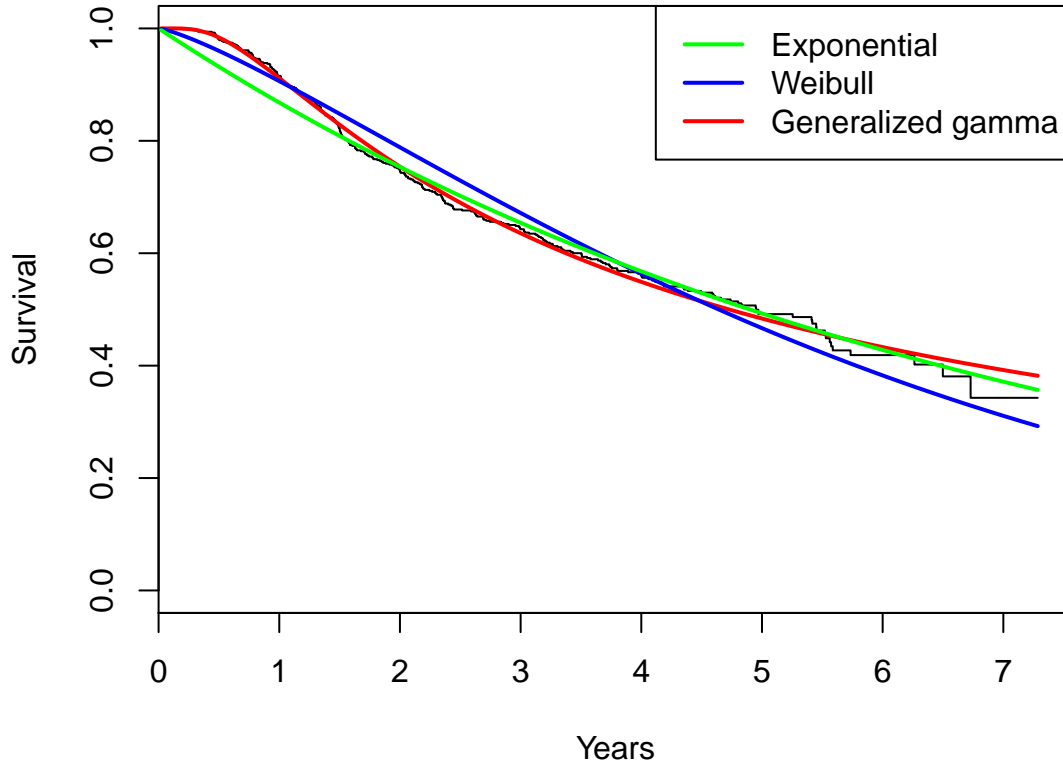
## Warning in seq.default(along = x$dlist$pars): partial argument match of 'along'
## to 'along.with'
## Warning in rep(col, length = nrow(X)): partial argument match of 'length' to
## 'length.out'
```

```
## Warning in rep(lty, length = nrow(X)): partial argument match of 'length' to
'length.out'
## Warning in rep(lwd, length = nrow(X)): partial argument match of 'length' to
'length.out'
## Warning in rep(col.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'
## Warning in rep(lty.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'
## Warning in rep(lwd.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'

lines(ex, col="green", ci=FALSE)

## Warning in seq.default(along = x$dlist$pars): partial argument match of 'along'
to 'along.with'
## Warning in rep(col, length = nrow(X)): partial argument match of 'length' to
'length.out'
## Warning in rep(lty, length = nrow(X)): partial argument match of 'length' to
'length.out'
## Warning in rep(lwd, length = nrow(X)): partial argument match of 'length' to
'length.out'
## Warning in rep(col.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'
## Warning in rep(lty.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'
## Warning in rep(lwd.ci, length = nrow(X)): partial argument match of 'length'
to 'length.out'

legend("topright", lty=c(1,1,1), lwd=c(2,2,2),
      col=c("green", "blue", "red"),
      c("Exponential", "Weibull", "Generalized gamma"))
```



Each model is a generalisation of the previous one, as described in the **flexsurv** documentation. Therefore we can use focused model comparison, with the “wide” model taken to be the generalized gamma, and assess if using a simpler model leads to improvements in precision of the estimates that outweigh any bias.

## 2. Focused model comparison and its challenges in this example

Focused model comparison requires all models being compared to be *nested* within a single “wide” model. This means that we can produce each submodel by fixing some of the parameters of the wide model at special values. This can be demonstrated for this example by implementing the Weibull and exponential models as generalized gamma models with parameters  $\mu, \sigma, Q$  fixed to special values. This is done by supplying **inits** and **fixedpars** to **flexsurvreg** as follows.

- the exponential has  $\sigma = 1, Q = 1$  and its rate parameter is  $1/\exp(\mu)$ , and is fitted as

```
ex2 <- flexsurvreg(Surv(reccrs, censrec) ~ 1, data=bc, dist="gengamma",
  inits=c(1,1,1), fixedpars=c(2,3))
```

- the Weibull model has  $Q = 1$ , shape  $1/\sigma$  and scale  $\exp(\mu)$ , and is fitted as

```
we2 <- flexsurvreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="gengamma",
  inits=c(1,1,1), fixedpars=3)
```

We can check that the parameter estimates returned by `ex2` and `we2` can be transformed easily to the estimates given by `ex` and `we`:

```
1 / exp(ex2$res["mu","est"])
1 / we2$res["sigma","est"]
exp(we2$est["mu","est"])
```

We now use `fic` to compare how well the three models estimate a focus parameter, which is taken to be the expected survival over 8 years (the *restricted mean* survival). When using `fic` for `flexsurvreg` models, the parameters `par` of the focus function should be on the log scale for parameters which are defined to be positive, in this example, for  $\sigma$ . Therefore, the focus is specified as a function of  $(\theta, \gamma) = (\mu, \log(\sigma), Q)$  instead of  $(\mu, \sigma, Q)$ .

```
focus <- function(par){
  rmst_gengamma(8, par[1], exp(par[2]), par[3])
}
```

The matrix `indmat` indicates which of the three parameters  $\mu, \sigma, Q$  are included in each model.

```
indmat <- rbind(exp = c(1,0,0),
  weib = c(1,1,0),
  ggamma = c(1,1,1))
```

Finally, in this example we also need to define the special values  $\gamma_0$  of the parameters  $\gamma = (\log(\sigma), Q)$  which define the “narrow” exponential model as a special case of the “wide” generalized gamma, as  $\gamma_0 = (0, 1)$ . Typically, e.g. in covariate selection problems,  $\gamma_0$  does not need to be defined, and would default to all 0.

```
gamma0 <- c(0,1)
```

Focused model comparison can now be performed. The generalized gamma model gives the estimate of 8-year survival with the lowest mean square error.

```
library(fic)
fic(gg, inds=indmat, gamma0=gamma0, focus=focus, sub=list(ex2, we2, gg))
```

##	vals	mods	rmse	rmse.adj	bias	se	FIC	focus
## 1	A	exp	0.439	0.439	-0.428	0.0985	125.51	4.79
## 2	A	weib	0.261	0.261	-0.230	0.1230	44.57	4.71
## 3	A	ggamma	0.128	0.128	0.000	0.1275	8.99	4.84

```

fic(gg, inds=indmat, gamma0=gamma0, focus=focus, sub=list(ex, we, gg))

##   vals   mods  rmse rmse.adj   bias    se    FIC focus
## 1    A    exp 0.439   0.439 -0.428 0.0985 125.51 0.141
## 2    A   weib 0.261   0.261 -0.230 0.1230  44.57 2.547
## 3    A ggamm 0.128   0.128  0.000 0.1275   8.99 4.844

```

Note that if we had supplied `sub=list(ex, we, gg)`, the wrong `focus` estimates would have been returned in the `fic` output for the exponential and Weibull models. This is because the models `ex` and `we` are not parameterised by  $\mu$  and  $\sigma$  as defined for the wide model, but by *transformations* of  $\mu$  and  $\sigma$ .

Alternatively we could have avoided fitting the exponential and Weibull models, and omitted the `sub` argument to `fic`. But it is usually sensible when comparing models to compare the models' estimates as well as their goodness-of-fit or adequacy. In this case, estimates of restricted mean survival over 8 years vary within about 0.2 years (4.7 to 4.9) between the three models.

We might expect a simpler model to give more precise estimates in situations with less data. In the following example, 50 uncensored survival times are simulated from a standard exponential distribution. A generalized gamma model is fitted to it, and treated as the “wide” model in a focused comparison of an exponential, Weibull and gamma.

```

set.seed(1)
y <- rexp(50); cen <- rep(1,50)
gge <- flexsurvreg(Surv(y, cen) ~ 1, dist="gengamma")
fic(gge, inds=indmat, gamma0=gamma0, focus=focus)

##   vals   mods  rmse rmse.adj   bias    se    FIC
## 1    A    exp 0.123   0.123 0.0177 0.121 0.0157
## 2    A   weib 0.122   0.122 0.0000 0.122 0.0146
## 3    A ggamm 0.122   0.122 0.0000 0.122 0.0152

```

In this case, the Weibull, rather than the generalized gamma, has the lowest MSE for the survival estimate, though the differences are tiny. Note that the focused analysis assumes that the biggest model is true, so it will not necessarily select the true model in situations where one exists, like in this artificial situation.

### 3. More complex parametric models

There is also a generalized F distribution implemented in **flexsurv**, which generalizes these models further by including a fourth parameter  $P$ . However the focused method will not be able to include this model in the comparison, as the special value  $P = 0$ , which defines the generalized gamma as a special case of the generalized F, is on the boundary of the parameter space, violating the asymptotic theory required by the method.

Spline-based models are an alternative way of defining very flexible parametric survival models, and can be fitted using the `flexsurvspline` function. Focused model comparison may be

possible for these models, however they would need to be set up carefully so that that smaller models are all nested within a single wide model, for example by choosing knot locations manually, and this has not been investigated.

See the main `fic` package vignette for focused comparison of standard Cox proportional hazards regression models.

#### 4. Focused comparison for survival models fitted with “`survreg`”

`fic` also has a built-in method for comparing parametric survival models fitted using the `survreg` function of the **survival** package (Therneau 2015). The exponential and Weibull models above can also be compared in the same way, but this time using the Weibull as the “wide” model. The generalized gamma is not included in `survreg`.

```
if (!require("survival"))
  stop("The `survival` package should be installed
to run code in this vignette")
ex <- survreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="exponential")

## Warning in seq.default(along = temp): partial argument match of 'along' to
'along.with'
## Warning: partial match of 'coef' to 'coefficients'

we <- survreg(Surv(recyrs, censrec) ~ 1, data=bc, dist="weibull")

## Warning in seq.default(along = temp): partial argument match of 'along' to
'along.with'
## Warning in seq.default(along = temp): partial match of 'coef' to 'coefficients'

indmat <- rbind(exp = c(1,0),
               weib = c(1,1))
```

The focus function is again specified as the restricted mean survival over 8 years, using the `rmst_weibull` function in the **flexsurv** package. An awkward reparametrisation is necessary to evaluate this **flexsurv**-based focus function at the parameters of the (`survreg`-based) models.

- The first parameter of the Weibull distribution in `flexsurvreg` equals  $1/\exp(\text{par}[2])$  where `par[2]` is the `Log(scale)` reported by `survreg`, as can be seen in `summary(we)`.
- The second parameter of the Weibull distribution in `flexsurvreg` equals  $\exp(\text{par}[1])$ , the exponential of the (`Intercept`) parameter reported in the `survreg` summary output.

As usual, the `par` argument of the focus function describes parameters on the real-line scale, that is, with any positive-valued parameters log transformed.

```
focus <- function(par){
  rmst_weibull(8, 1/exp(par[2]), exp(par[1]))
}
```

Finally, `fic` shows that the more flexible Weibull gives a more precise estimate with lower RMSE, as we found before.

```
fic(we, inds=indmat, focus=focus, sub=list(ex, we))
```

##	vals	mods	rmse	rmse.adj	bias	se	FIC	focus
## 1	A	exp	0.144	0.144	0.0744	0.123	3.797	4.79
## 2	A	weib	0.124	0.124	0.0000	0.124	0.326	4.71

## References

- Jackson C (2016). “flexsurv: A Platform for Parametric Survival Modeling in R.” *Journal of Statistical Software*, **70**(8), 1–33. doi:[10.18637/jss.v070.i08](https://doi.org/10.18637/jss.v070.i08).
- Sauerbrei W, Royston P (1999). “Building Multivariable Prognostic and Diagnostic Models: Transformation of the Predictors by Using Fractional Polynomials.” *Journal of the Royal Statistical Society A*, **162**(1), 71–94.
- Therneau TM (2015). *A Package for Survival Analysis in S*. Version 2.38, URL <https://CRAN.R-project.org/package=survival>.