

Parametric Survival Regression

David M. Rocke

November 16, 28, 2023

Exponential Regression

For each subject i define a linear predictor

$$\begin{aligned}\eta_i &= \beta_1 x_{i1} + \cdots + \beta_p x_{ip} \\ h_i(t|\text{covariates}) &= \lambda e^{\eta_i} = \lambda \theta_i\end{aligned}$$

This has a log link as in a generalized linear model. Since the hazard does not depend on t , the hazards are (trivially) proportional. Larger values of η mean larger hazard, so earlier failure times.

Accelerated Failure Time

Suppose that $S_i(t) = S_0(t\theta_i)$ where $\theta_i = \exp(\eta_i)$ and $\eta_i = \beta_1 x_{i1} + \dots + \beta_p x_{ip}$. This is called an accelerated failure time model because covariates cause uniform acceleration (or slowing) of failure times. If the base distribution is exponential with parameter λ then

$$S_i(t) = e^{-\lambda\theta_i t}$$

which is an exponential model with base hazard multiplied by θ_i , which is also the proportional hazards model. If η is larger, then this corresponds to larger hazard and earlier failure times as before.

Accelerated Failure Time

In terms of the log survival time, $Y = \ln(T)$ for an exponential distribution with parameter λ can be written as

$$Y = \alpha + W$$
$$\alpha = -\ln(\lambda)$$

where W has the Gumbel extreme value distribution. Since the failure rate of the regression model is $-\lambda\theta_i$, in the representation of the distribution of $Y = \ln(T)$, the term $\alpha = -\ln(\lambda)$ would be replaced by

$$-\ln(\lambda\theta_i) = -\ln(\lambda) - \ln(\theta_i) = \alpha - \eta_i$$

Accelerated Failure Time

In the software we usually use, the the model is written as

$$Y = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + \cdots + \beta_p x_{pi} + W$$
$$\alpha = -\ln(\lambda)$$

so that larger values of η_i imply larger values of Y , and thus later failure times. Thus, coefficients in this model will be of the opposite sign of those for coxph.

Accelerated Failure Time

For a Weibull distribution, the hazard function and the survival function are

$$\begin{aligned}h(t) &= \lambda p(\lambda t)^{p-1} \\S(t) &= e^{-(\lambda t)^p}\end{aligned}$$

We can construct a proportional hazards model by using a linear predictor η_i without constant term and letting $\theta_i = e^{\eta_i}$ we have

$$h(t) = \lambda p(\lambda t)^{p-1} \theta_i$$

Accelerated Failure Time

A distribution with $h(t) = \lambda p(\lambda t)^{p-1} \theta_i$ is a Weibull distribution with parameters $\lambda^* = \lambda \theta_i^{1/p}$ and p so the survival function is

$$\begin{aligned} S^*(t) &= e^{-(\lambda^* t)^p} \\ &= e^{-(\lambda \theta_i^{1/p} t)^p} \\ &= S(t \theta_i^{1/p}) \end{aligned}$$

so this is also an accelerated failure time model.

Accelerated Failure Time

In terms of the log survival time $Y = \ln(T)$ the model can be written as

$$Y = \alpha - \sigma\eta + \sigma W$$

$$\alpha = -\ln(\lambda)$$

$$\sigma = 1/p$$

where W has the extreme value distribution. The estimated parameter λ is the intercept and the other coefficients are those of $-\eta$, which will be typically of the opposite sign of those for coxph.

Accelerated Failure Time

These AFT models are log-linear, meaning that the linear predictor has a log link. The exponential and the Weibull are the only log-linear models that are simultaneously proportional hazards models. Other parametric distributions can be used for survival regression either as a proportional hazards model or as an accelerated failure time model.

survreg {survival} R Documentation

Regression for a Parametric Survival Model

Description

Fit a parametric survival regression model.

These are location-scale models for an arbitrary transform of the time variable; the most common cases use a log transformation, leading to accelerated failure time models.

Usage

```
survreg(formula, data, weights, subset,  
        na.action, dist="weibull", init=NULL, scale=0,  
        control, parms=NULL, model=FALSE, x=FALSE,  
        y=TRUE, robust=FALSE, cluster, score=FALSE, ...)
```

Arguments

formula

a formula expression as for other regression models. The response is usually a survival object as returned by the `Surv` function.

See the documentation for `Surv`, `lm` and `formula` for details.

data

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

dist

assumed distribution for y variable. If the argument is a character string, then it is assumed to name an element from `survreg.distributions`. These include "weibull", "exponential", "gaussian", "logistic", "lognormal", and "loglogistic". Otherwise, it is assumed to be a user defined list conforming to the format described in `survreg.distributions`.

parms

a list of fixed parameters. For the t-distribution for instance this is the degrees of freedom; most of the distributions have no parameters.

scale

optional fixed value for the scale. If set to ≤ 0 then the scale is estimated.

```
> names(survreg.distributions)
"extreme"      "logistic"    "gaussian"    "weibull"    "exponential"
"rayleigh"    "loggaussian" "lognormal"   "loglogistic" "t"
```

```

> anderson.cox0 <- coxph(anderson.surv~treat,data=anderson)
> summary(anderson.cox0)
Call:
coxph(formula = anderson.surv ~ treat, data = anderson)

n= 42, number of events= 30

              coef exp(coef) se(coef)      z Pr(>|z|)
treatstandard 1.5721    4.8169  0.4124 3.812 0.000138 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
treatstandard    4.817    0.2076    2.147    10.81

Concordance= 0.69 (se = 0.041 )
Likelihood ratio test= 16.35 on 1 df,  p=5e-05
Wald test            = 14.53 on 1 df,  p=1e-04
Score (logrank) test = 17.25 on 1 df,  p=3e-05

```

```
> anderson.weib <- survreg(anderson.surv~treat,data=anderson)
> summary(anderson.weib)
```

Call:

```
survreg(formula = anderson.surv ~ treat, data = anderson)
```

	Value	Std. Error	z	p
(Intercept)	3.516	0.252	13.96	< 2e-16
treatstandard	-1.267	0.311	-4.08	4.5e-05
Log(scale)	-0.312	0.147	-2.12	0.034

Scale= 0.732

Weibull distribution

Loglik(model)= -106.6 Loglik(intercept only)= -116.4

Chisq= 19.65 on 1 degrees of freedom, p= 9.3e-06

Number of Newton-Raphson Iterations: 5

n= 42

```
> anderson.exp <- survreg(anderson.surv~treat,data=anderson,dist="exp")
> summary(anderson.exp)
```

Call:

```
survreg(formula = anderson.surv ~ treat, data = anderson, dist = "exp")
```

	Value	Std. Error	z	p
(Intercept)	3.686	0.333	11.06	< 2e-16
treatstandard	-1.527	0.398	-3.83	0.00013

Scale fixed at 1

Exponential distribution

Loglik(model)= -108.5 Loglik(intercept only)= -116.8

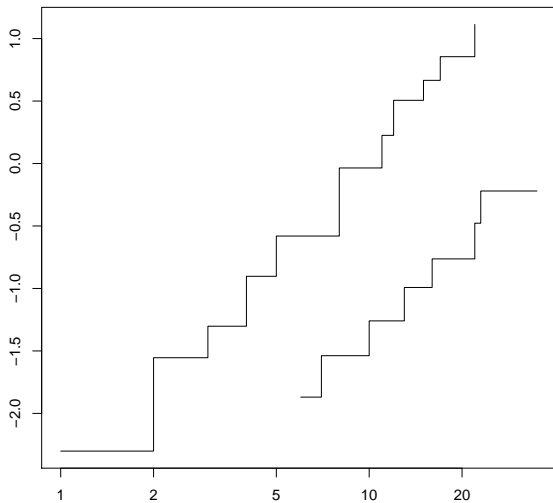
Chisq= 16.49 on 1 degrees of freedom, p= 4.9e-05

Number of Newton-Raphson Iterations: 4

n= 42

```
> plot(survfit(anderson.surv~treat,data=anderson),fun="cloglog")
```

If the cloglog plot survfit is linear, then a Weibull model may be ok.



In terms of the tests for the coefficient and model, we have

Model	Coef(treat)	z-score	p-value	LRT	p-value
Cox	1.5721	3.812	1.38×10^{-4}	16.35	5×10^{-5}
Weibull	-1.267	-4.08	5×10^{-5}	19.65	9.3×10^{-6}
Exponential	-1.527	-3.83	1.3×10^{-4}	16.49	4.9×10^{-5}

The results are similar in the three cases, with the exponential being closer to the Cox model than the Weibull is. We can examine the evidence that the true scale parameter is 1:

```
> anova(anderson.weib, anderson.exp, test="Chisq")
  Terms Resid. Df    -2*LL Test Df  Deviance Pr(>Chi)
1 treat      39 213.1590      NA      NA      NA
2 treat      40 217.0481    = -1 -3.889116  0.0486
```

The likelihood ratio statistics is -3.889 with one df, and the χ^2 significance level is $p = 0.0486$, which is borderline.

KM Larynx Data—Case Study

Kardaun (1983) reports data on 90 males diagnosed with cancer of the larynx during the period 1970–1978 at a Dutch hospital. Times recorded are the intervals (in years) between first treatment and either death or the end of the study (January 1, 1983). Also recorded are the patient's age at the time of diagnosis, the year of diagnosis, and the stage of the patient's cancer.

KM Larynx Data—Case Study

The four stages of disease in the study were based on the T.N.M. (primary tumor (T), nodal involvement (N) and distant metastasis (M) grading) classification used by the American Joint Committee for Cancer Staging (1972). The four groups are Stage I, T1N0M0 with 33 patients; Stage II, T2N0M0 with 17 patients; Stage III, T3N0M0 and T_xN1M0, with 27 patients; $x = 1, 2, \text{ or } 3$; and Stage IV, all other TNM combinations except T1S with 13 patients. The stages are ordered from least serious to most serious.

KM larynx data from Section 1.8

Description

The larynx data frame has 90 rows and 5 columns.

Format

This data frame contains the following columns:

stage	Stage of disease (1=stage 1, 2=stage2, 3=stage 3, 4=stage 4)
time	Time to death or on-study time, years (erroneously given as months)
age	Age at diagnosis of larynx cancer
diagyr	Year of diagnosis of larynx cancer
delta	Death indicator (0=alive, 1=dead)

Source

Klein and Moeschberger (1997) Survival Analysis Techniques for
Censored and Truncated data, Springer.
Kardaun, Stat. Nederlandica 37 (1983), 103-126.

```

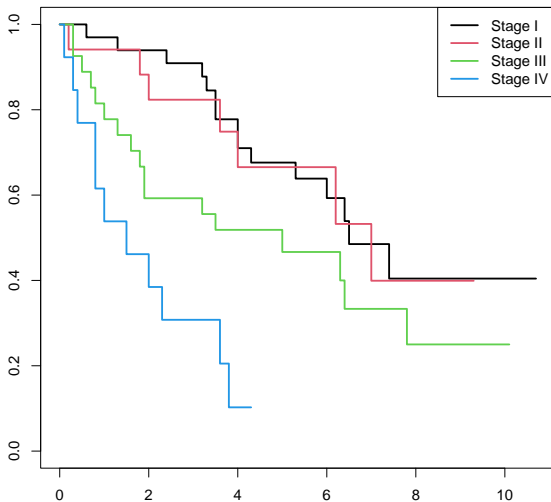
library(survival)
library(KMsurv)
data(larynx)
larynx1 <- larynx
larynx1$stage <- factor(larynx1$stage,
                        labels=c("Stage I","Stage II","Stage III","Stage IV"))
larynx1.surv <- with(larynx1,Surv(time,delta))

plotL1 <- function(){
  pdf("larynxSurvival")
  plot(survfit(larynx1.surv~stage,data=larynx1),col=1:4,lwd=2)
  legend("topright",c("Stage I","Stage II","Stage III","Stage IV"),col=1:4,lwd=2)
  title("Survival Curves for Four Stages of Larynx Data")
  dev.off()
}

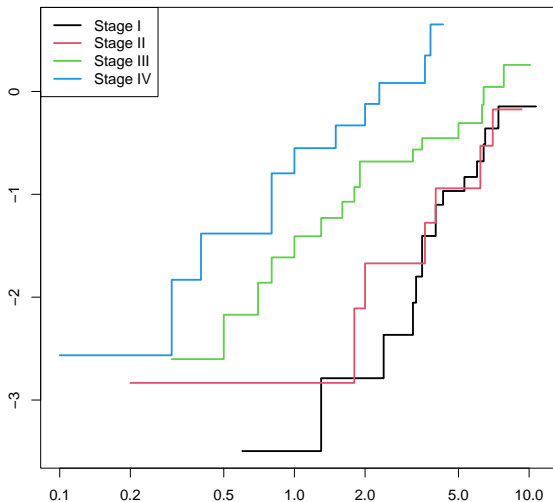
plotL2 <- function(){
  pdf("larynxCloglog.pdf")
  plot(survfit(larynx1.surv~stage,data=larynx1,type="fleming"),col=1:4,lwd=2,
       fun="cloglog")
  legend("topleft",c("Stage I","Stage II","Stage III","Stage IV"),col=1:4,lwd=2)
  title("C-LogLog Curves for Four Stages of Larynx Data")
  dev.off()
}

```

Survival Curves for Four Stages of Larynx Data



C-LogLog Curves for Four Stages of Larynx Data



```
larynx1.cox1 <- coxph(larynx1.surv~stage*age+diagyr,data=larynx1)
print(summary(larynx1.cox1))
```

Call:

```
coxph(formula = larynx1.surv ~ stage * age + diagyr, data = larynx1)
```

```
n= 90, number of events= 50
```

	coef	exp(coef)	se(coef)	z	Pr(> z)
stageStage II	-8.1188354	0.0002979	3.7766491	-2.150	0.0316 *
stageStage III	-0.1703272	0.8433888	2.4775437	-0.069	0.9452
stageStage IV	0.8169275	2.2635344	2.4291035	0.336	0.7366
age	-0.0029151	0.9970891	0.0260894	-0.112	0.9110
diagyr	0.0035098	1.0035159	0.0755288	0.046	0.9629
stageStage II:age	0.1228516	1.1307166	0.0536604	2.289	0.0221 *
stageStage III:age	0.0121220	1.0121957	0.0375822	0.323	0.7470
stageStage IV:age	0.0142748	1.0143771	0.0359427	0.397	0.6913

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

	exp(coef)	exp(-coef)	lower .95	upper .95
stageStage II	0.0002979	3357.1086	1.817e-07	0.4883
stageStage III	0.8433888	1.1857	6.564e-03	108.3719
stageStage IV	2.2635344	0.4418	1.937e-02	264.5110
age	0.9970891	1.0029	9.474e-01	1.0494
diagyr	1.0035159	0.9965	8.654e-01	1.1636
stageStage II:age	1.1307166	0.8844	1.018e+00	1.2561
stageStage III:age	1.0121957	0.9880	9.403e-01	1.0896
stageStage IV:age	1.0143771	0.9858	9.454e-01	1.0884

Concordance= 0.69 (se = 0.038)

Likelihood ratio test= 24.67 on 8 df, p=0.002

Wald test = 24.47 on 8 df, p=0.002

Score (logrank) test = 29.17 on 8 df, p=3e-04

Call:

```
coxph(formula = larynx1.surv ~ stage + age + diagyr, data = larynx1)
```

```
n= 90, number of events= 50
```

	coef	exp(coef)	se(coef)	z	Pr(> z)
stageStage II	0.15164	1.16375	0.46481	0.326	0.7442
stageStage III	0.64473	1.90546	0.35619	1.810	0.0703 .
stageStage IV	1.73211	5.65255	0.43596	3.973	7.09e-05 ***
age	0.01869	1.01887	0.01433	1.304	0.1922
diagyr	-0.01819	0.98198	0.07646	-0.238	0.8120

```
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

	exp(coef)	exp(-coef)	lower .95	upper .95
stageStage II	1.164	0.8593	0.4680	2.894
stageStage III	1.905	0.5248	0.9480	3.830
stageStage IV	5.653	0.1769	2.4052	13.284
age	1.019	0.9815	0.9906	1.048
diagyr	0.982	1.0184	0.8453	1.141

```
Concordance= 0.674 (se = 0.039 )
```

```
Likelihood ratio test= 18.37 on 5 df, p=0.003
```

```
Wald test = 21.2 on 5 df, p=7e-04
```

```
Score (logrank) test = 24.84 on 5 df, p=1e-04
```

```
> print(drop1(larynx1.cox2,test="Chisq"))
```

```
Single term deletions
```

```
Model:
```

```
larynx1.surv ~ stage + age + diagyr
```

	Df	AIC	LRT	Pr(>Chi)	
<none>		385.36			
stage	3	394.87	15.5153	0.001425	**
age	1	385.10	1.7412	0.186988	
diagyr	1	383.41	0.0565	0.812188	

```
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Stage is clearly important. Age and year of diagnosis are not yet shown to be useful.

Schoenfeld Residuals

```
> print(cox.zph(larynx1.cox2))  
      chisq df    p  
stage   3.72  3 0.29  
age     1.16  1 0.28  
diagyr  1.24  1 0.27  
GLOBAL  6.68  5 0.25
```

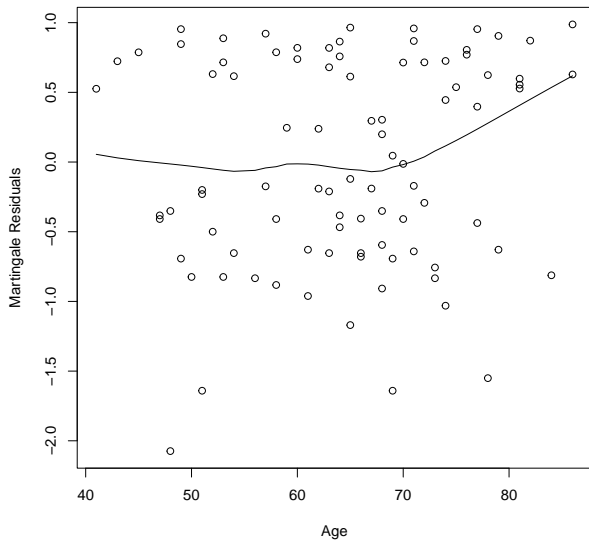
No significant correlation of residuals with time suggesting no evidence of lack of proportionality. This is consistent with the CLogLog plot shown earlier.

Martingale Residuals

```
plotLM.age <- function(){
  pdf("larynxLMage.pdf")
  mres <- residuals(coxph(larynx1.surv~stage+diagyr,data=larynx1),type="martingale")
  plot(larynx1$age,mres,xlab="Age",ylab="Martingale Residuals")
  lines(lowess(larynx1$age,mres))
  title("Martingale Residuals vs. Age")
  dev.off()
}

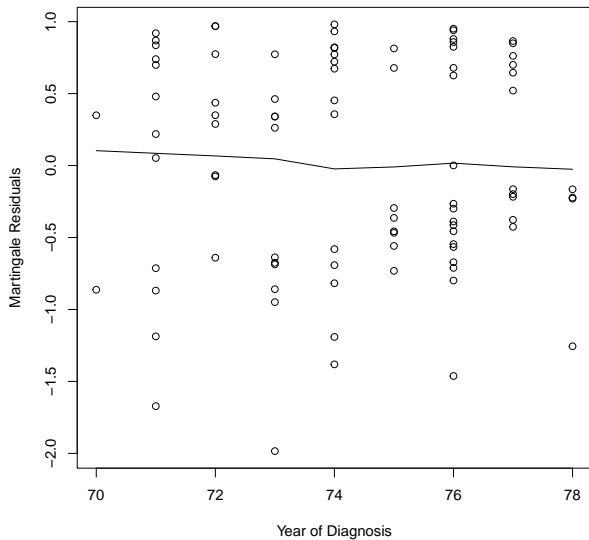
plotLM.diagyr <- function(){
  pdf("larynxLMdiagyr.pdf")
  mres <- residuals(coxph(larynx1.surv~stage+age,data=larynx1),type="martingale")
  plot(larynx1$diagyr,mres,xlab="Year of Diagnosis",ylab="Martingale Residuals")
  lines(lowess(larynx1$diagyr,mres))
  title("Martingale Residuals vs. Year of Diagnosis")
  dev.off()
}
```

Martingale Residuals vs. Age



Possibly age matters
between under 75 and
at least 75 at diagnosis.

Martingale Residuals vs. Year of Diagnosis



This variable is not significant and no transformation is suggested.

Revised Model

```
larynx2 <- data.frame(larynx1,age75=factor(larynx1$age >= 75,
      labels=c("Under 75","75 or Older")))
larynx2.surv <- with(larynx2,Surv(time,delta))
larynx2.cox1 <- coxph(larynx2.surv~stage+age75,data=larynx2)
print(summary(larynx2.cox1))
```

Call:

```
coxph(formula = larynx2.surv ~ stage + age75, data = larynx2)
```

n= 90, number of events= 50

	coef	exp(coef)	se(coef)	z	Pr(> z)	
stageStage II	0.2098	1.2335	0.4659	0.450	0.6524	
stageStage III	0.6859	1.9856	0.3579	1.917	0.0553	.
stageStage IV	1.7409	5.7023	0.4187	4.158	3.21e-05	***
age7575 or Older	0.7064	2.0268	0.3225	2.190	0.0285	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
stageStage II	1.233	0.8107	0.4950	3.074
stageStage III	1.986	0.5036	0.9846	4.004
stageStage IV	5.702	0.1754	2.5101	12.955
age7575 or Older	2.027	0.4934	1.0771	3.814

Concordance= 0.695 (se = 0.035)

Likelihood ratio test= 20.81 on 4 df, p=3e-04

Wald test = 24.31 on 4 df, p=7e-05

Score (logrank) test = 28.03 on 4 df, p=1e-05

```
> drop1(larynx2.cox1,test="Chisq")
```

Single term deletions

Model:

```
larynx2.surv ~ stage + age75
```

	Df	AIC	LRT	Pr(>Chi)	
<none>		380.92			
stage	3	391.24	16.3248	0.0009727	***
age75	1	383.24	4.3218	0.0376276	*

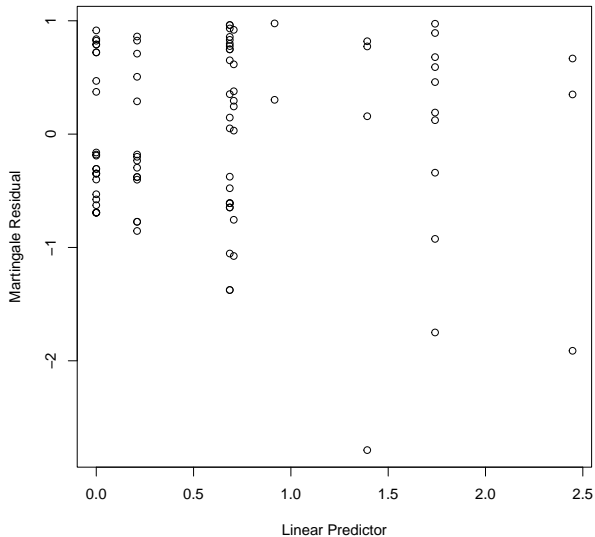
 Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


```
larynx2.mart <- residuals(larynx2.cox1,type="martingale")
larynx2.dev <- residuals(larynx2.cox1,type="deviance")
larynx2.dfb <- residuals(larynx2.cox1,type="dfbeta")
larynx2.preds <- predict(larynx2.cox1) #linear predictor

plotrp1 <- function(){
  pdf("plotrp1.pdf")
  plot(larynx2.preds,larynx2.mart,xlab="Linear Predictor",ylab="Martingale Residual")
  title("Martingale Residuals vs. Linear Predictor")
  dev.off()
}

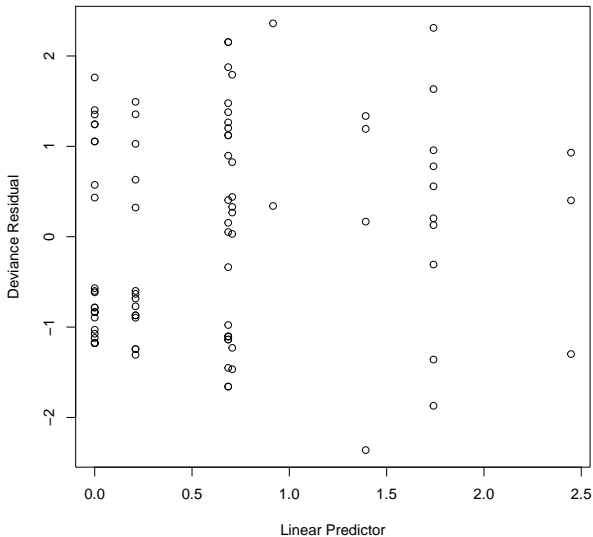
plotrp2 <- function(){
  pdf("plotrp2.pdf")
  plot(larynx2.preds,larynx2.dev,xlab="Linear Predictor",ylab="Deviance Residual")
  title("Deviance Residuals vs. Linear Predictor")
  dev.off()
}
```

Martingale Residuals vs. Linear Predictor



The small martingale residuals are patients 75 or over, censored, with short times.

Deviance Residuals vs. Linear Predictor



Weibull Regression

```
larynx2.weib <- survreg(larynx2.surv~stage+age75,data=larynx2)
print(summary(larynx2.cox1))
print(summary(larynx2.weib))
coxph(formula = larynx2.surv ~ stage + age75, data = larynx2)
```

	coef	exp(coef)	se(coef)	z	Pr(> z)
stageStage II	0.2098	1.2335	0.4659	0.450	0.6524
stageStage III	0.6859	1.9856	0.3579	1.917	0.0553 .
stageStage IV	1.7409	5.7023	0.4187	4.158	3.21e-05 ***
age7575 or Older	0.7064	2.0268	0.3225	2.190	0.0285 *

```
survreg(formula = larynx2.surv ~ stage + age75, data = larynx2)
```

	Value	Std. Error	z	p
(Intercept)	2.552	0.266	9.58	< 2e-16
stageStage II	-0.225	0.409	-0.55	0.583
stageStage III	-0.632	0.321	-1.97	0.049
stageStage IV	-1.580	0.361	-4.37	1.2e-05
age7575 or Older	-0.644	0.291	-2.21	0.027
Log(scale)	-0.130	0.122	-1.07	0.286

Scale= 0.878

	exp(coef)	exp(-coef)	lower .95	upper .95
stageStage II	1.233	0.8107	0.4950	3.074
stageStage III	1.986	0.5036	0.9846	4.004
stageStage IV	5.702	0.1754	2.5101	12.955
age7575 or Older	2.027	0.4934	1.0771	3.814

Concordance= 0.695 (se = 0.035)

Likelihood ratio test= 20.81 on 4 df, p=3e-04

Wald test = 24.31 on 4 df, p=7e-05

Score (logrank) test = 28.03 on 4 df, p=1e-05

Weibull distribution

Loglik(model)= -140.1 Loglik(intercept only)= -151.1

Chisq= 22.03 on 4 degrees of freedom, p= 2e-04

Number of Newton-Raphson Iterations: 5

n= 90

```
> drop1(larynx2.cox1,test="Chisq")
Single term deletions
```

Model:

```
larynx2.surv ~ stage + age75
      Df      AIC      LRT Pr(>Chi)
<none>    380.92
stage    3 391.24 16.3248 0.0009727 ***
age75    1 383.24  4.3218 0.0376276 *
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```
> drop1(larynx2.weib,test="Chisq")
Single term deletions
```

Model:

```
larynx2.surv ~ stage + age75
      Df      AIC      LRT Pr(>Chi)
<none>    292.19
stage    3 303.69 17.499 0.0005579 ***
age75    1 294.82  4.633 0.0313627 *
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

In this case, the Weibull model fits well and the power is likely slightly higher. The p-values in most of the tests are smaller, and the confidence intervals are tighter. This does not always happen, and mostly what we can see from the two approaches is that the conclusions are similar.

Residual Analysis for Parametric Models

```
residuals.survreg {survival}
Compute Residuals for 'survreg' Objects
```

```
residuals(object, type=c("response", "deviance", "dfbeta", "dfbetas",
  "working", "ldcase", "ldresp", "ldshape", "matrix"), rsigma=TRUE,
  collapse=FALSE, weighted=FALSE, ...)
```

Arguments

`object`

an object inheriting from class `survreg`.

`type`

type of residuals, with choices of "response", "deviance", "dfbeta", "dfbetas", "working", "ldcase", "lsresp", "ldshape", and "matrix".

Response residuals are on the scale of the original data, working residuals are on the scale of the linear predictor, and deviance residuals are on the log-likelihood scale

Residual Analysis for Parametric Models

```
predict.survreg {survival}
Predicted Values for a 'survreg' Object
```

```
predict(object, newdata,
  type=c("response", "link", "lp", "linear", "terms", "quantile", "uquantile"),
  se.fit=FALSE, terms=NULL, p=c(0.1, 0.9), na.action=na.pass, ...)
```

Arguments

object

result of a model fit using the `survreg` function.

newdata

data for prediction. If absent, predictions are for the subjects used in the original fit.

type

the type of predicted value. This can be on the original scale of the data (`response`), the linear predictor (`linear`, with `lp` as an allowed abbreviation), a predicted quantile on the original scale of the data (`quantile`), a quantile on the linear predictor scale (`uquantile`), or the matrix of terms for the linear predictor (`terms`). At this time `link` and linear predictor (`lp`) are identical.

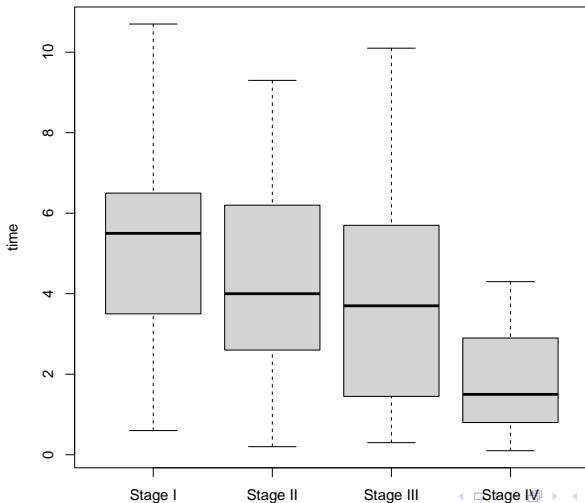
Residual Analysis for Parametric Models

Many of the plots are similar to those for non-parametric survival analysis. Point predictions are more meaningful here because of the parametric pdf, cdf, survival curve, and hazard function, though predictions come with wide intervals (even for the exponential). Some useful plots:

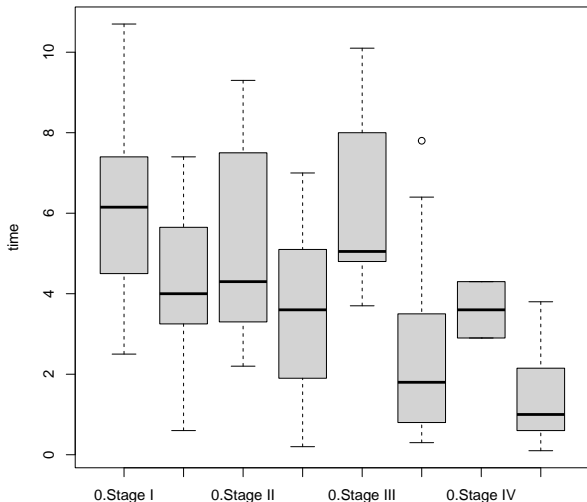
- 1 Working residuals (linear predictor scale) vs. linear predictor.
- 2 Deviance residuals vs. linear predictor.
- 3 Dfbeta plots vs. observation order for all coefficients.

In each case we may want to identify and characterize the unusual observations.

What is an Unusual Time?



What is an Unusual Time?



For stage I, event times less than 2 years are unusual. For stage IV, event times greater than 4 years are unusual.

All event times are between 0.1 years and 7.8 years.

All censoring times are between 2.2 years and 10.7 years.

```
larynx2.weib <- survreg(larynx2.surv~stage+age75,data=larynx2)

larynx2w.work <- residuals(larynx2.weib,type="working")
larynx2w.dev <- residuals(larynx2.weib,type="deviance")
larynx2w.dfb <- residuals(larynx2.weib,type="dfbetas")
larynx2w.preds <- predict(larynx2.weib,type="lp") #linear predictor

larynx.boxplot1 <- function(){
  pdf("boxplotw1.pdf")
  with(larynx2,boxplot(time~stage))
  dev.off()
}

larynx.boxplot2 <- function(){
  pdf("boxplotw2.pdf")
  with(larynx2,boxplot(time~delta+stage))
  dev.off()
}
```

```

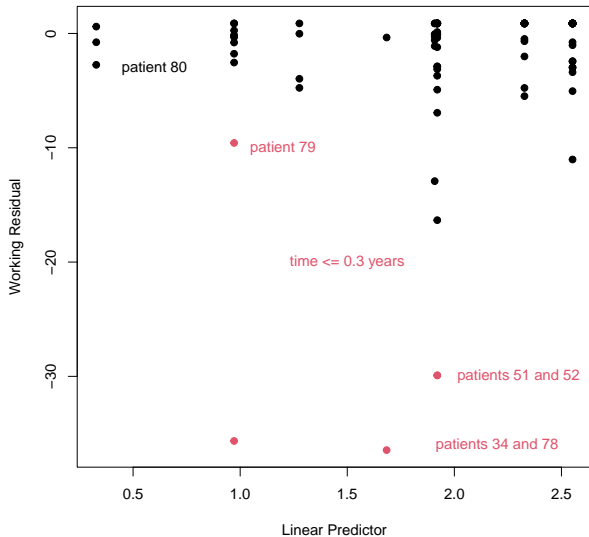
plotwr1 <- function(){
  pdf("plotwrp1.pdf")
  plot(larynx2w.preds,larynx2w.work,xlab="Linear Predictor",
       ylab="Working Residual",col=(as.numeric(larynx2$time <= .3)+1),pch=19)
  text(1.5,-20,"time <= 0.3 years",col=2)
  text(2.3,-30,"patients 51 and 52",col=2)
  text(2.2,-36,"patients 34 and 78",col=2)
  text(1.2,-10,"patient 79",col=2)
  text(0.6,-3,"patient 80")
  title("Working Residuals vs. Linear Predictor")
  dev.off()
}

plotwr2 <- function(){
  pdf("plotwrp2.pdf")
  plot(larynx2w.preds,larynx2w.dev,xlab="Linear Predictor",ylab="Deviance Residual")
  title("Deviance Residuals vs. Linear Predictor")
  dev.off()
}

plotwrdfb1 <- function(){
  pdf("plotwrdfb1.pdf")
  plot(larynx2w.dfb[,1],xlab="Observation Order",ylab="dfbeta Intercept")
  text(70,-.19, "75",col=3)
  text(82,-.17,"89",col=3)
  title("dfbeta Values by Observation Order")
  dev.off()
}

```

Working Residuals vs. Linear Predictor



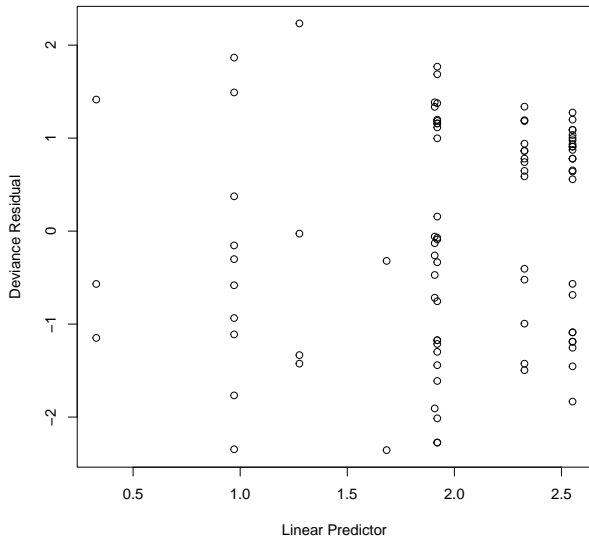
The four most unusual points have times less than or equal to 0.3 years.

```
print(larynx2[larynx2$time <= 0.3,])
  stage time age diagyr delta      age75
34 Stage II  0.2  86     74     1 75 or Older
51 Stage III 0.3  49     72     1   Under 75
52 Stage III 0.3  71     76     1   Under 75
78 Stage IV  0.1  65     72     1   Under 75
79 Stage IV  0.3  71     76     1   Under 75

> with(larynx2,sort(time[stage=="Stage IV" & delta==1]))
[1] 0.1 0.3 0.4 0.8 0.8 1.0 1.5 2.0 2.3 3.6 3.8
```

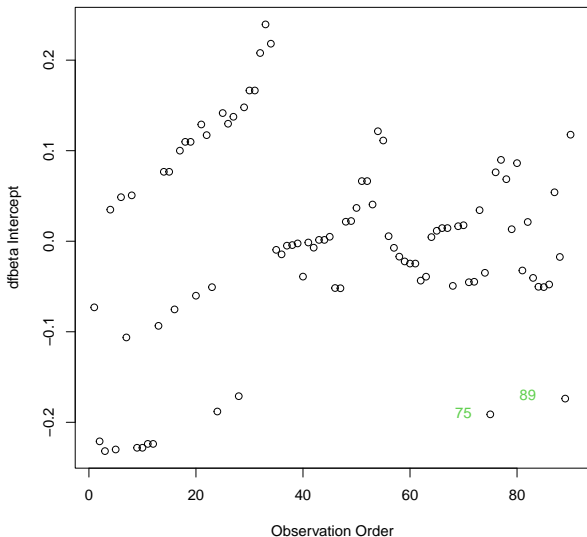
Patients 51 and 52 have the same predicted value (stage/age) and the same time, so the same residual. Their event time is small for stage III. Patients 34 and 78 have the two lowest times, and 0.1 is low even for stage IV. Patient 79 has the second smallest event time for stage IV, but closer to the third smallest time, which is patient 80, stage 4, age 76.

Deviance Residuals vs. Linear Predictor



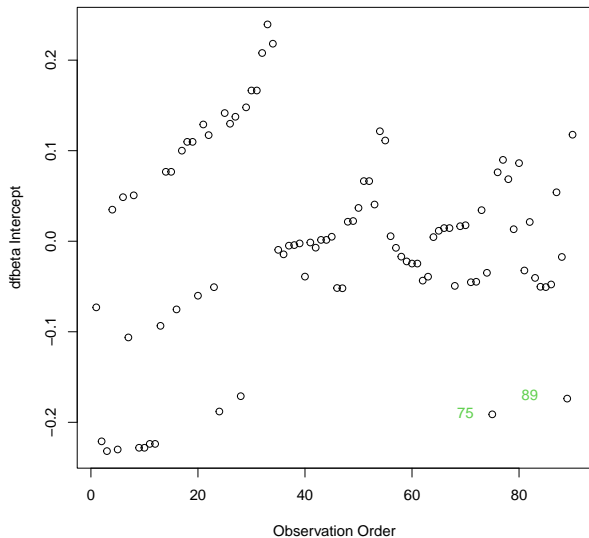
No really unusual points.

dfbeta Values by Observation Order



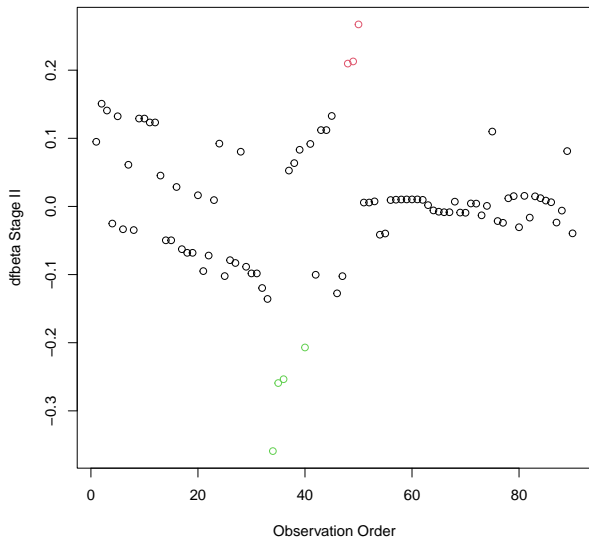
Dfbeta for the intercept. The data are sorted by stage, then by time within stage, which explains the straight or curved lines. Patient 75 had the third longest time and in stage III, censored, while over age 75. The two longer times were also censored and under age 75.

dfbeta Values by Observation Order



Patient 89 is stage IV and over age 75 with an event time of 3.8 years, largest in that group and second largest time in stage IV.

dfbeta Values by Observation Order

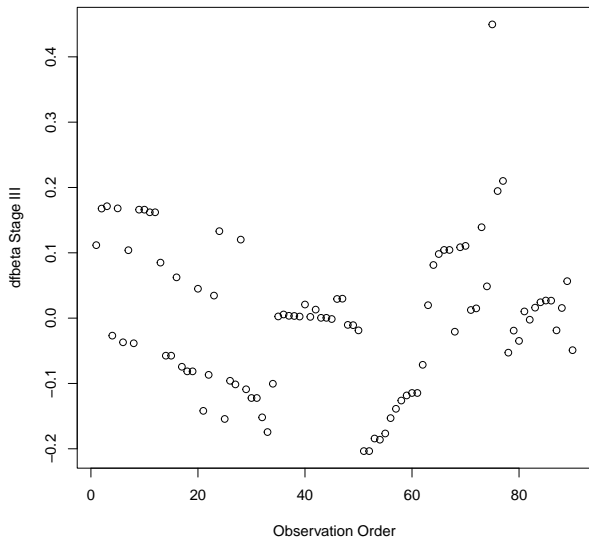


Dfbeta for stage II vs. stage I.

Patients 48–50 (red) are stage II with high times, censored.

Patients 34–36, 40 (green) are stage II with low event times.

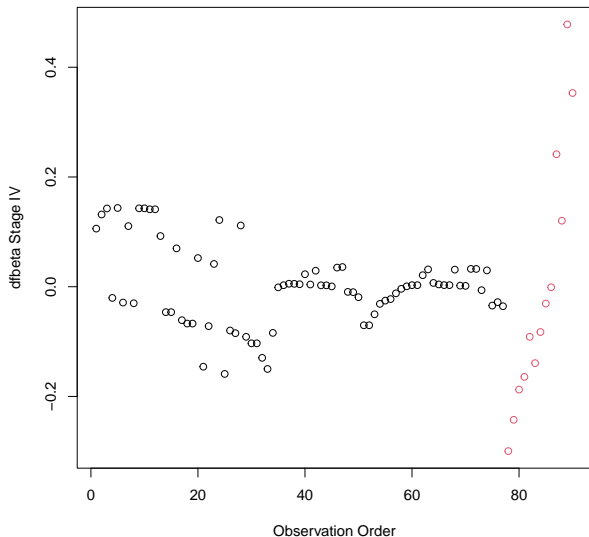
dfbeta Values by Observation Order



Dfbeta for stage III vs. stage I.

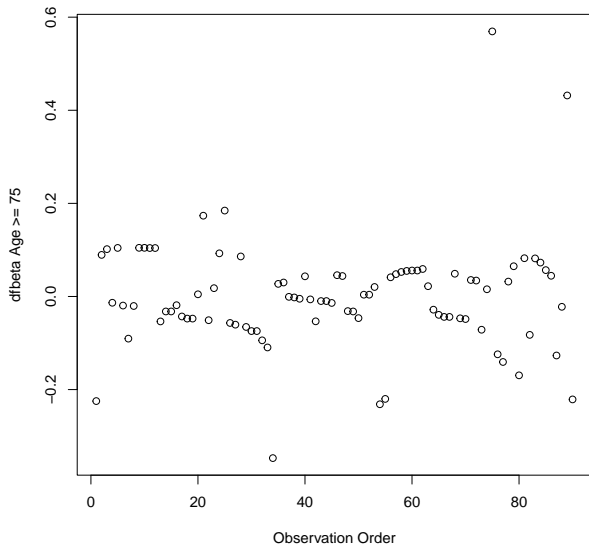
The outlying point at the top is patient 75 who is over age 75 and had a time, censored, of 8 years, highest among that group. Patient 75 is also influential for the intercept.

dfbeta Values by Observation Order



Dfbeta for stage IV (red) vs. stage I. The three points in the upper right are patients 89, 90, and 87 from the most extreme down. Patient 89 is age 84 and had the largest event time in stage IV (also influential for intercept). Patients 90 and 87 (under age 75) have the two largest censored times in stage IV

dfbeta Values by Observation Order



Dfbeta for age ≥ 75 . The two largest points are patients 75, and 89, also influential for the intercept (both), stage III (patient 75) and stage IV (patient 89). Patient 75, age 78, is stage III with a high time, censored. Patient 89, age 84, is stage IV with a high event time.

Choice of Distribution

There are many specific distributions that have been used for survival analysis. The Weibull, which we have been using for the `larynx` data, is perhaps the most common, but there are many others. We will now consider a number of possible choices for these data and for methods to compare the results and to choose which one to use. We will use the package `flexsurv`, which fits all these models in a consistent fashion.

Choice of Distribution

The distributions below are mostly generalized gamma distributions.

Exponential \subset Weibull \subset GenGamma

Gamma \subset GenGamma

Lognormal \subset GenGamma

Log Logistic

We can test hypotheses via the coefficients or via the likelihood ratio test for nested models. We can also choose based on the AIC. We will also fit the log logistic, which is not nested in any of these.

Distribution	Parameters			larynx df
GenGamma	μ	σ	Q	7
Weibull	$\text{scale} = \exp(\mu)$	$\text{shape} = 1/\sigma$	$Q = 1$	6
Exponential	$\text{scale} = 1$ ($\mu = 0$)	$\text{shape} = 1/\sigma$	$Q = 1$	5
Lognormal	μ	σ	$Q = 0$	6
Gamma	$\text{rate} = \exp(-\mu)/\sigma^2$	$\text{shape} = 1/\sigma^2$	$Q = \sigma$	6
Log Logistic	scale	shape	—	6

```
flexsurvreg(formula = larynx2.surv ~ stage + age75, data = larynx2,
  dist = "gengamma")
```

Estimates:

	data	mean	est	L95%	U95%	se	exp(est)	L95%	U95%
mu	NA		2.4774	1.8954	3.0594	0.2970	NA	NA	NA
sigma	NA		1.1363	0.7752	1.6657	0.2217	NA	NA	NA
Q	NA		0.3086	-0.7456	1.3629	0.5379	NA	NA	NA
stageStage II	0.1889		-0.3003	-1.1561	0.5555	0.4367	0.7406	0.3147	1.7429
stageStage III	0.3000		-0.8895	-1.6739	-0.1051	0.4002	0.4109	0.1875	0.9002
stageStage IV	0.1444		-1.8368	-2.7232	-0.9504	0.4523	0.1593	0.0657	0.3866
age7575 or Older	0.1889		-0.7908	-1.4820	-0.0996	0.3527	0.4535	0.2272	0.9052

N = 90, Events: 50, Censored: 40

Total time at risk: 377.8

Log-likelihood = -139.324, df = 7

AIC = 292.6479

The parameter Q has 95% CI $(-0.7456, 1.3629)$. Thus the lognormal ($Q = 0$) and the Weibull/Exponential ($Q = 1$) are not excluded. We can make a table of the LLR tests and the AIC as additional resources.

	Distribution	df	AIC	Log.Likelihood	Test	Statistic	p.value
1	GenGamma	7	292.6479	-139.3240	---	NA	NA
2	Weibull	6	292.1862	-140.0931	GenGamma	1.5382381	0.2148803
3	Exponential	5	291.2664	-140.6332	Weibull	1.0802717	0.2986368
4	Exponential	5	291.2664	-140.6332	GenGamma	2.6185098	0.2700212
5	Log Normal	6	290.9645	-139.4822	GenGamma	0.3165648	0.5736794
6	Gamma	6	291.8330	-139.9165	GenGamma	1.1851073	0.2763187
7	Log Logistic	6	291.2727	-139.6364	---	NA	NA

None of the tests is significant, meaning that we can discard the GenGamma, but it is not clear from the tests which is to be preferred. The lowest AIC is for the log normal, but the exponential and log logistic are also viable. In this case, the choice probably does not matter much and might be left to other considerations. This might include what choices are usual in the area of study.

Covariate Coefficients

	GenGamma	Weibull	Exponential	Log Normal	Gamma	Log Logistic
stageStage II	-0.3002770	-0.2245329	0.2356561	-0.3410281	0.2326337	-0.2688519
stageStage III	-0.8895095	-0.6316799	0.7054481	-0.9961833	0.6595373	-0.9058758
stageStage IV	-1.8368331	-1.5801957	1.6860437	-1.9361069	1.6060334	-1.8212578
age7575 or Older	-0.7908116	-0.6435282	0.7214757	-0.8438686	0.6566930	-0.7735850

Z-Scores

	GenGamma	Weibull	Exponential	Log Normal	Gamma	Log Logistic
stageStage II	-0.687680	-0.5489061	0.5063924	-0.782943	0.5727488	-0.647444
stageStage III	-2.222585	-1.9687502	1.9739091	-2.765596	2.0884299	-2.577930
stageStage IV	-4.061451	-4.3724738	4.2470390	-4.463763	4.5027893	-4.403049
age7575 or Older	-2.242377	-2.2082025	2.2401154	-2.376887	2.2642766	-2.271082

The covariate coefficients and their significance also does not change much among the distributions, though the signs do differ for the exponential and gamma vs. the others.