S-Plus Instructions
Parametric regression with survreg( )

Suppose that you have the following regression model for some continuous, positive random variable T and a vector of explanatory variables (covariates) $\boldsymbol{x}$:

$$\ln T = b_0 + b'\mathbf{x} + s W,$$

where W is a continuous random variable on the real line with a distribution that does not involve any unknown parameters. Choices of distributions of T that can be fitted with survreg( ) are extreme value, logistic, gaussian (normal) and Rayleigh, corresponding to, respectively, Weibull, log-logistic, log-normal and log-Rayleigh distribution for T. S-Plus calls $\hat{a}_0$ (Intercept), *ó* Dispersion(scale) and the beta's in front of a regressor by the name given to the regressor.

General syntax of survreg( )

```
USAGE:
 survreg(formula = formula(data), subset, link = c("log"),
      dist = c("extreme", "logistic", "gaussian", "exponential",
      "rayleigh"), fixed = list())
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.
subset:   expression saying that only a subset of the rows of the
       data should be used in the fit.
link:     transformation to be used on the y variable.
dist:     assumed distribution for the transformed y variable.
fixed:    a list of fixed parameters, most often just the scale.
```

(Note: there are other optional arguments for survreg( ) … see S-Plus help files.)

Example

Let us now fit a log-logistic regression model to the data of Assignment 1, Question 11, assuming no left truncation.

```
> a1q11 <- read.table("a1q11.dat",header=T)
> a1q11.t <- a1q11[,1]
> a1q11.d <- a1q11[,2]
> a1q11.x <- as.numeric(a1q11[,3]=="D")
> a1q11.sr <- survreg(Surv(a1q11.t,a1q11.d)~a1q11.x,dist="logistic")
>a1q11.srD <-survreg(Surv(a1q11.t[a1q11.x==1],a1q11.d[a1q11.x==1]),
+ dist="logistic")
> a1q11.tD <- a1q11.t[a1q11.x==1]
> a1q11.dD <- a1q11.d[a1q11.x==1]
> a1q11.srD <- survreg(Surv(a1q11.t[a1q11.x==1],a1q11.d[a1q11.x==1])~1,
+ dist="logistic")
> a1q11.srP <- survreg(Surv(a1q11.t[a1q11.x==0],a1q11.d[a1q11.x==0])~1,
+ dist="logistic")
```

Let us now look at excerpts of the output of survreg( )

```
> summary(a1q11.sr)

Coefficients:
            Value Std. Error z value     p
(Intercept) 3.312      0.124  26.743 0.000
    a1q11.x 0.148      0.163   0.908 0.364

Logistic distribution: Dispersion (scale) est = 0.2126292
Degrees of Freedom: 31 Total; 28 Residual
-2*Log-Likelihood: 21.4

> summary(a1q11.srD)

Coefficients:
            Value Std. Error z value p
(Intercept)  3.46      0.113    30.5 0

Logistic distribution: Dispersion (scale) est = 0.2247572
Degrees of Freedom: 17 Total; 15 Residual
-2*Log-Likelihood: 12.8

> summary(a1q11.srP)

Coefficients:
            Value Std. Error z value p
(Intercept)   3.3      0.121    27.3 0

Logistic distribution: Dispersion (scale) est = 0.1956445
Degrees of Freedom: 14 Total; 12 Residual
-2*Log-Likelihood: 8.41
```

If we wish to test whether it is reasonable to assume that the scale parameter does not depend on the value of the covariate, we simply fit a model with two different scale parameters (2^{nd} and 3^{rd} survreg( )) and one model with a common scale parameter (the 1^{st} survreg( )) and we test if the model reduction is appropriate using a likelihood ratio test. Here, the likelihood ratio statistic is 21.4-(12.8+8.41) = 0.19, for a p-value of 0.66.