ACCELERATED FAILURE TIME MODEL

Dr. Aric LaBarr
Institute for Advanced Analytics

MODEL STRUCTURE

Accelerated Failure Time Model

- We can transform this model into a linear regression model by taking the natural log of both sides of the equation:

$$T_i = e^{\beta_0+\beta_1 x_{i,1}+\cdots+\beta_k x_{i,k}+\sigma e_i}$$

- The equation now becomes:

$$\log T_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_k x_{i,k} + \sigma e_i$$

AFT Model – R

```
recid.aft.ln <- survreg(Surv(week, arrest == 1) ~
                        fin + age + race + wexp + mar + paro + prio,
                        data = recid, dist = 'lognormal')

summary(recid.aft.ln)
```

AFT Model – R

```
## Call:
## survreg(formula = Surv(week, arrest == 1) ~ fin + age + race +
##     wexp + mar + paro + prio, data = recid, dist = "lognormal")
##               Value Std. Error     z       p
## (Intercept)  4.2677     0.4617  9.24 < 2e-16
## fin          0.3428     0.1641  2.09 0.03667
## age          0.0272     0.0158  1.73 0.08427
## race        -0.3632     0.2647 -1.37 0.17006
## wexp         0.2681     0.1789  1.50 0.13391
## mar          0.4604     0.2951  1.56 0.11882
## paro         0.0559     0.1691  0.33 0.74108
## prio        -0.0655     0.0271 -2.42 0.01559
## Log(scale)   0.2582     0.0764  3.38 0.00073
##
## Scale= 1.29
##
## Log Normal distribution
## Loglik(model)= -683.2   Loglik(intercept only)= -697.9
##  Chisq= 29.35 on 7 degrees of freedom, p= 0.00012
## Number of Newton-Raphson Iterations: 4
## n= 432
```

?

ERROR DISTRIBUTIONS

Model Assumptions

Accelerated Failure Time Model

$$\log T_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_k x_{i,k} + \sigma e_i$$

Errors in the model

- The errors in the AFT model can follow many different distributions.
- Assumptions:
  - **Specify correct distribution of errors**
  - Constant Mean
  - Constant Variance (σ)
  - Independence across observations

Variance (Scale) vs. Rate

$$\log T_i = \beta_0 + \beta_1 x_{i,1} + \cdots + \beta_k x_{i,k} + \sigma e_i$$

Variance of the errors

- Variance (also called scale in survival analysis) describes the spread of the distribution of errors.
- Another common form is the inverse of the scale, called the **rate**: $\lambda = 1/\sigma$.
- If σ is small, then events are not spread out → events happening close to one another → higher rate of events, or λ is large.

Alternative Distributions

- We will focus on the distribution of failure time T (not on the error itself) since this is what we input into software.
- Distributions are commonly checked two ways:
  1. Graphically
  2. Statistical Tests

- We will go over some commonly used distributions for survival data, but there is **no guarantee** that your data will adequately match just one of the distributions here, or even any of them at all.

Matching up the parameterization

R	SAS	Parameter
	`proc lifereg` "Weibull Shape"	γ
`survreg` "scale"	`proc lifereg` "scale"	$1/\gamma$
`survreg` "intercept"	`proc lifereg` "intercept"	$-\log\lambda$

Exponential vs. Weibull – R

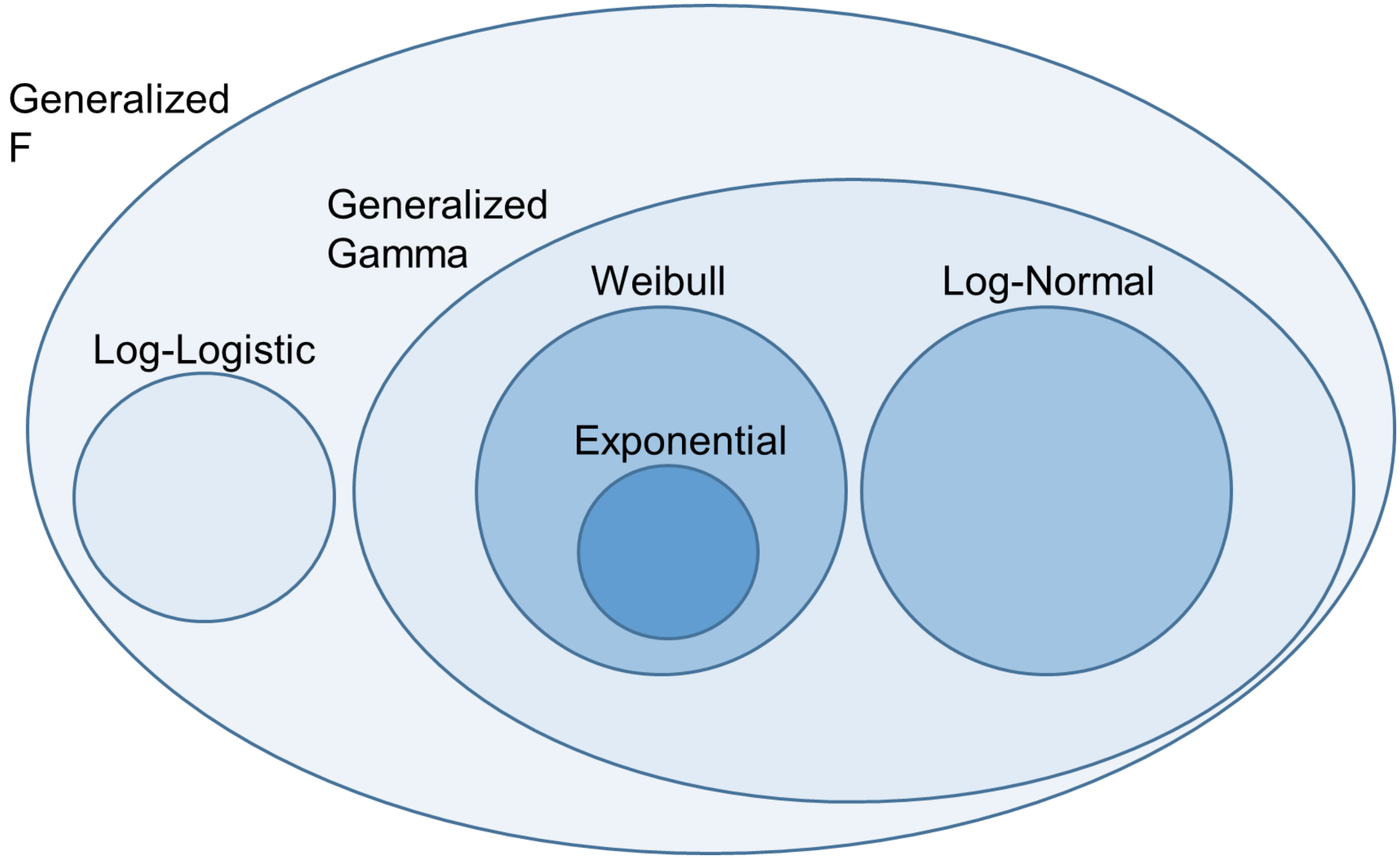
```
recid.aft.w <- survreg(Surv(week, arrest == 1) ~
                       fin + age + race + wexp + mar + paro + prio,
                       data = recid, dist = 'weibull')

summary(recid.aft.w)
```

Exponential vs. Weibull – R

```
## Call:
## survreg(formula = Surv(week, arrest == 1) ~ fin + age + race +
##     wexp + mar + paro + prio, data = recid, dist = "weibull")
##                  Value Std. Error     z       p
## (Intercept)  3.9901     0.4191  9.52 < 2e-16
## fin          0.2722     0.1380  1.97 0.04852
## age          0.0407     0.0160  2.54 0.01096
## race        -0.2248     0.2202 -1.02 0.30721
## wexp         0.1066     0.1515  0.70 0.48196
## mar          0.3113     0.2733  1.14 0.25473
## paro         0.0588     0.1396  0.42 0.67355
## prio        -0.0658     0.0209 -3.14 0.00167
## Log(scale)  -0.3391     0.0890 -3.81 0.00014
##
## Scale= 0.712
##
## Weibull distribution
## Loglik(model)= -679.9   Loglik(intercept only)= -696.6
##  Chisq= 33.42 on 7 degrees of freedom, p= 2.2e-05
## Number of Newton-Raphson Iterations: 6
## n= 432
```

Other Distributions

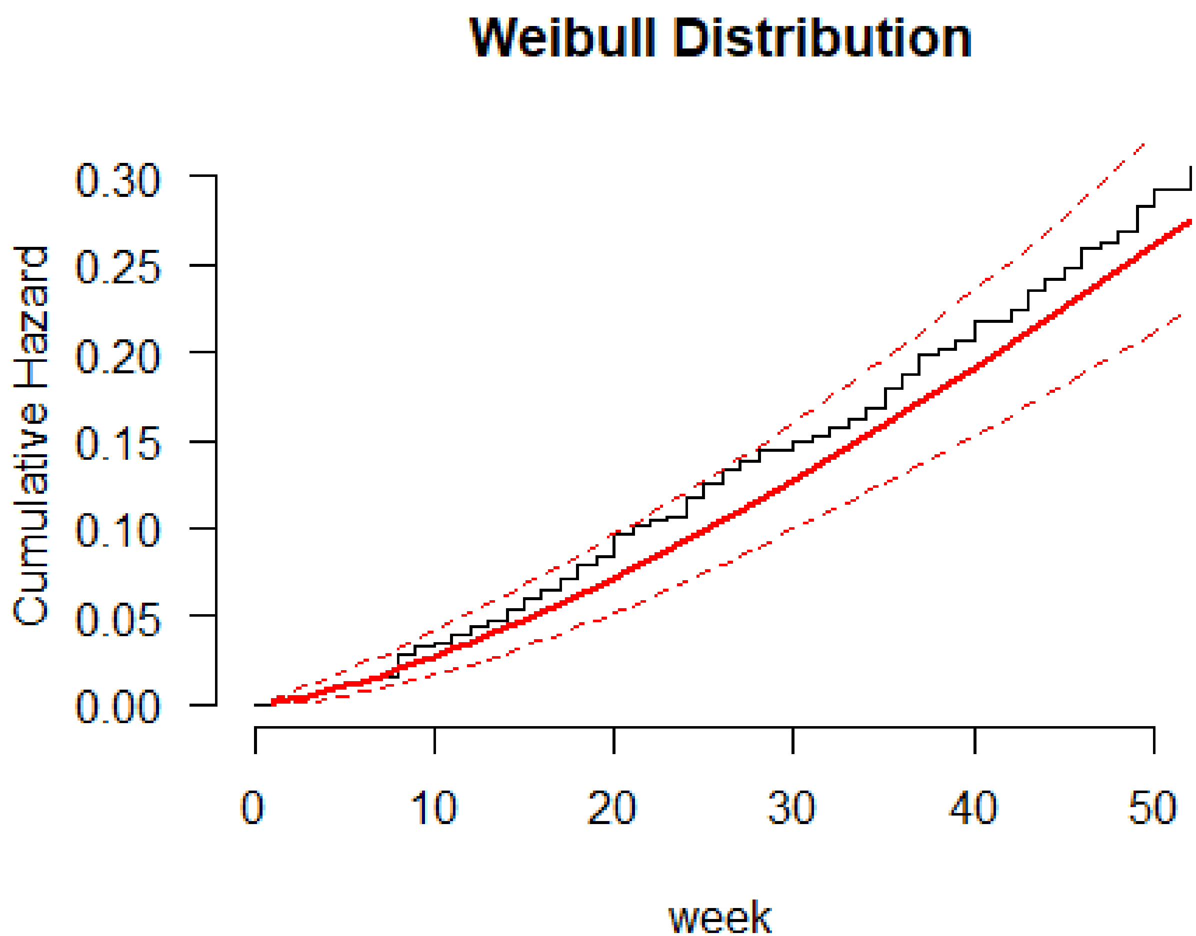
- **Generalized F Distribution:** Includes log-logistic and generalized gamma as special cases.



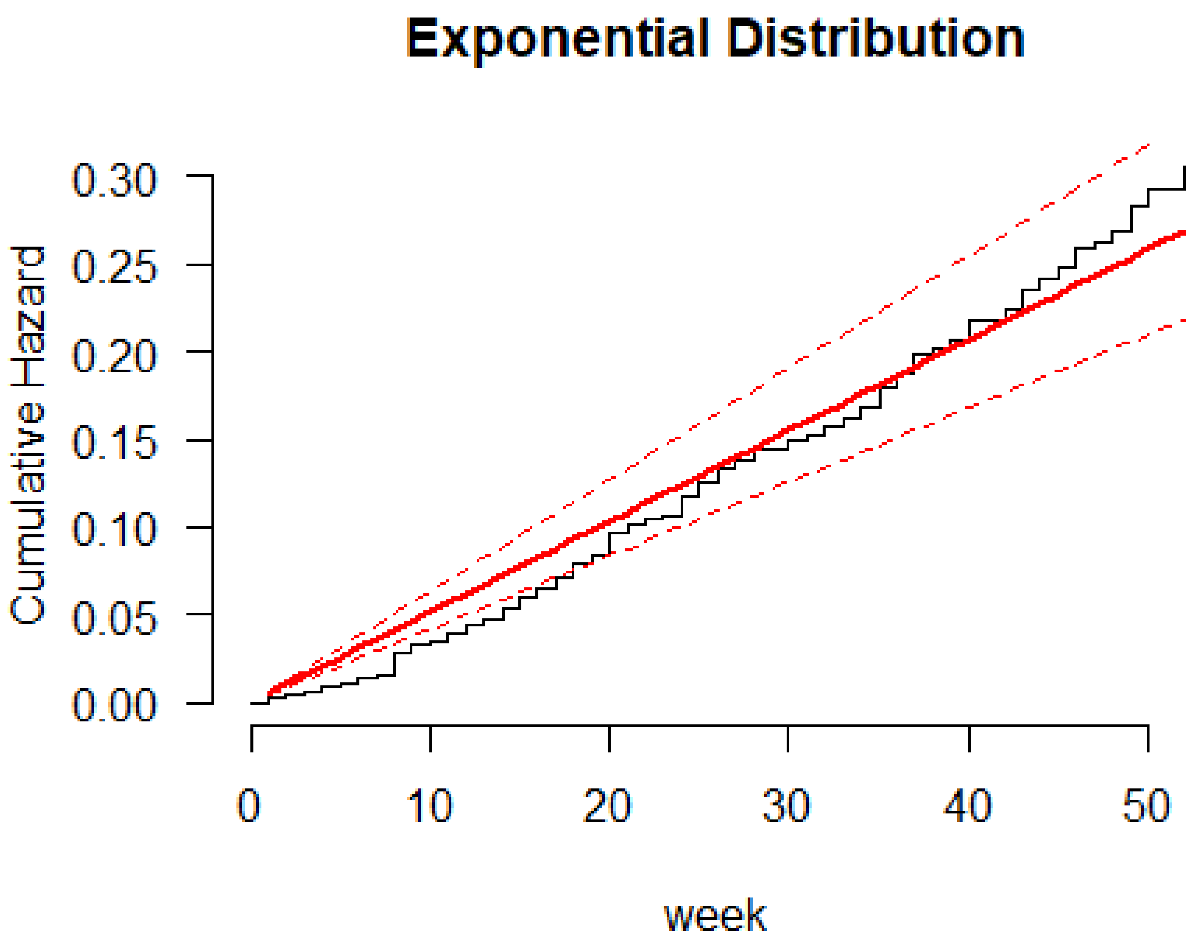
Checking Distributions – R

```
recid.aft.w <- flexsurvreg(Surv(week, arrest == 1) ~
                           fin + age + race + wexp +
                           mar + paro + prio,
                           data = recid, dist = "weibull")

plot(recid.aft.w, type = "cumhaz", ci = TRUE, conf.int = FALSE,
     las = 1, bty = "n", xlab = "week", ylab = "Cumulative Hazard",
     main = "Weibull Distribution")
```

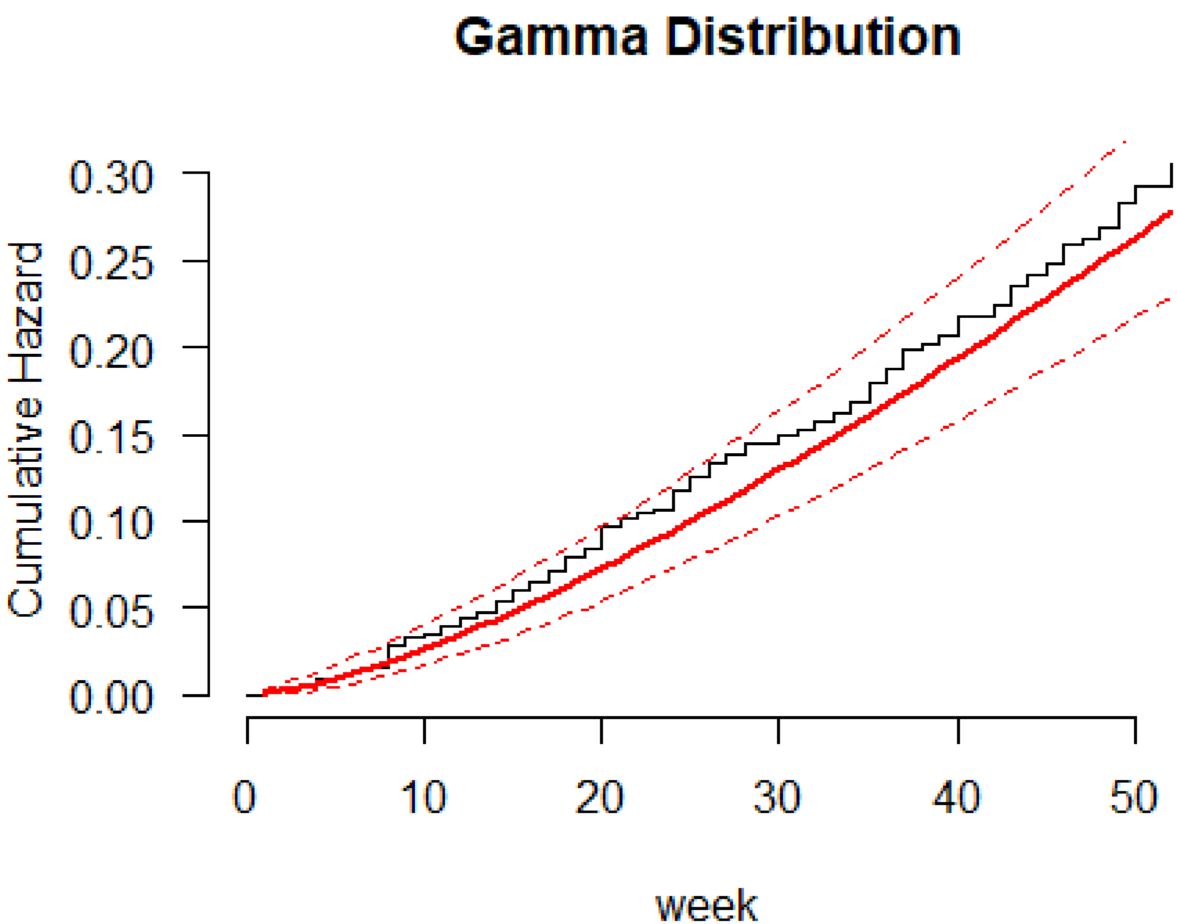
Checking Distributions – R



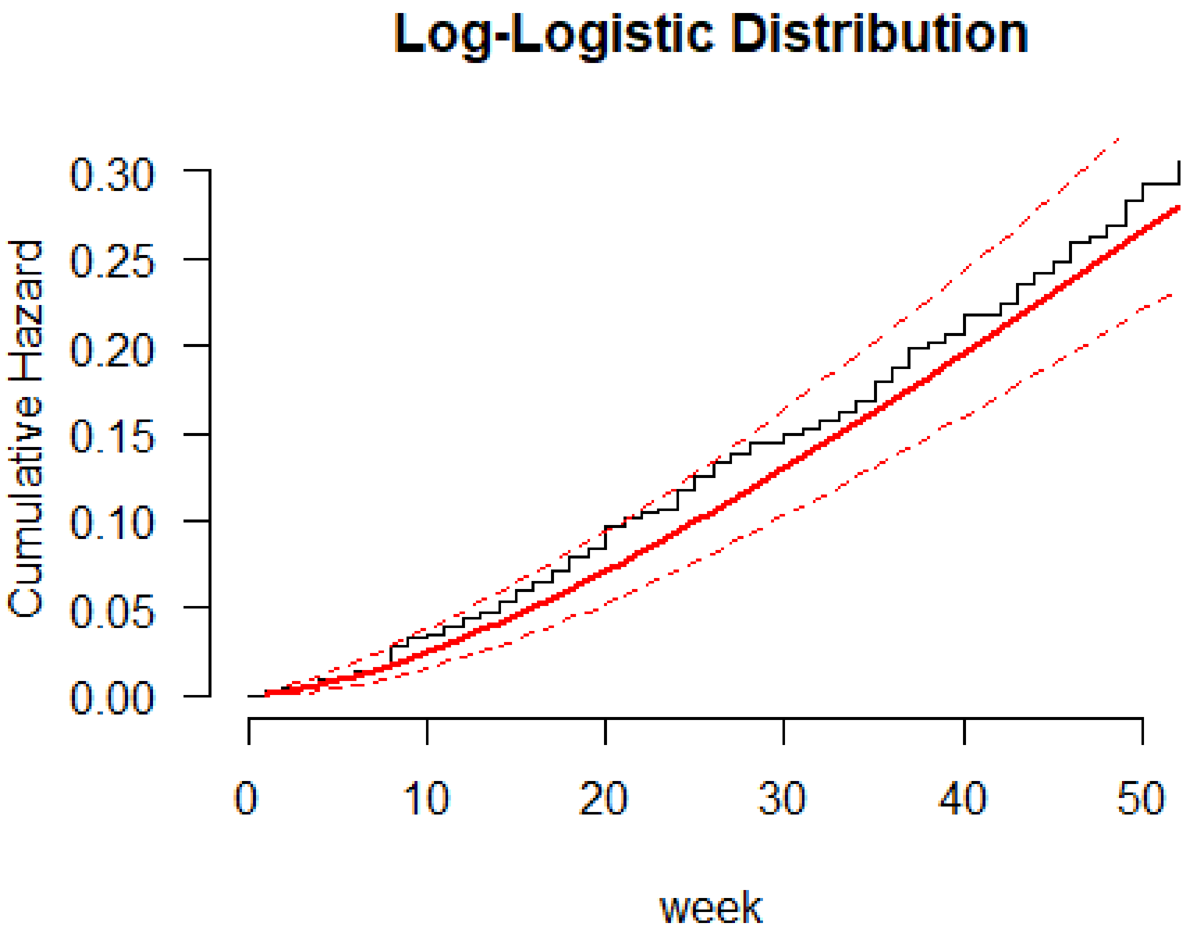
Checking Distributions – R



Checking Distributions – R



Checking Distributions – R



Goodness-of-Fit Tests

- Since these models are nested within the generalized gamma, we can use the **likelihood ratio test**.
- Likelihood Ratio Test:

$$\mathrm{LRT} = -2(\log L_{Nested} - \log L_{Full})$$

- Typically, use **full model** (all variables) since we don't know which p-values are correct.

Goodness-of-Fit Tests – R

```
like.e <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                data = recid, dist = "exp")$loglik
like.w <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                data = recid, dist = "weibull")$loglik
like.ln <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                data = recid, dist = "lnorm")$loglik
like.g <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                data = recid, dist = "gamma")$loglik
like.ll <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                data = recid, dist = "llogis")$loglik
like.f <- flexsurvreg(Surv(week, arrest == 1) ~
                         fin + age + race + wexp + mar + paro + prio,
                         data = recid, dist = "genf")$loglik
```

Goodness-of-Fit Tests – R

```
pval.e.g <- 1 - pchisq((-2*(like.e-like.g)), 2)
pval.w.g <- 1 - pchisq((-2*(like.w-like.g)), 1)
pval.ln.g <- 1 - pchisq((-2*(like.ln-like.g)), 1)
pval.g.f <- 1 - pchisq((-2*(like.g-like.f)), 1)
pval.ll.f <- 1 - pchisq((-2*(like.ll-like.f)), 1)

Tests <- c('Exp vs. Gam', 'Wei vs. Gam', 'LogN vs. Gam', 'Gam vs. F',
           'LogL vs. F')

P_values <- c(pval.e.g, pval.w.g, pval.ln.g, pval.g.f, pval.ll.f)

cbind(Tests, P_values)
```

Goodness-of-Fit Tests – R

```
##       Tests            P_values
## [1,] "Exp vs. Gam"   "0.00172559564523367"
## [2,] "Wei vs. Gam"   "1"
## [3,] "LogN vs. Gam" "0.0110221983305441"
## [4,] "Gam vs. F"     "0.108860911475402"
## [5,] "LogL vs. F"    "0.118276422245853"
```

PREDICTING SURVIVAL & EVENT TIMES

Making Predictions

- AFT models assume a distribution for T, meaning that we expect event times to behave in a certain way.

- **IF WE ASSUME CORRECT DISTRIBUTION** we can predict quantiles, survival probabilities, event times, survival curves, and changes in expected values as predictor variable values change.

Predicted Survival Quantiles – R

```
recid.aft.w <- survreg(Surv(week, arrest == 1) ~
                         fin + age + prio, data = recid,
                         dist = 'weibull')

survprob.75.50.25 <- predict(recid.aft.w, type = "quantile",
                               se.fit = TRUE,
                             p = c(0.25, 0.5, 0.75))

head(survprob.75.50.25$fit)
```

```
##          [,1]      [,2]      [,3]
## [1,] 52.68849  98.72758 161.95827
## [2,] 24.17956  45.30760  74.32514
## [3,] 17.89085  33.52383  54.99438
## [4,] 64.22717 120.34873 197.42682
## [5,] 35.95471  67.37185 110.52057
## [6,] 48.95457  91.73097 150.48064
```

Predicted (Mean) Event Times – R

```
p.time.mean <- predict(recid.aft.w, type = "response",
                       se.fit = TRUE)

head(p.time.mean$fit, n = 10)
```

```
##  [1] 128.26394  58.86229  43.55317 156.35349  87.52751
        119.17415 143.73152
##  [8] 115.26040  81.92984 113.19494
```

Predicted Survival Probability at *t* – R

```
survprob.actual <- 1 - psurvreg(recid$week,
                                mean = predict(recid.aft.w,
                                type = "lp"),
                                scale = recid.aft.w$scale,
                                distribution = recid.aft.w$dist)

head(survprob.actual, n = 10)
```

```
##  [1] 0.9285822 0.8389085 0.6315234 0.8073231 0.6173609
     0.7312118 0.9260438
##  [8] 0.7203354 0.5891529 0.7143008
```

Predicted Survival Probability at *t* – R

```
survprob.10wk <- 1 - psurvreg(10,
                              mean = predict(recid.aft.w,
                              type = "lp"),
                              scale = recid.aft.w$scale,
                              distribution = recid.aft.w$dist)

head(survprob.10wk)
```

```
## [1] 0.9723202 0.9198457 0.8803901 0.9789527 0.9531961 0.9693657
```

Predicted Change in Event Time – R

```
new_time <-  qsurvreg(1 - survprob.actual,
                      mean = predict(recid.aft.w, type = "lp") +
                      coef(recid.aft.w)['fin'],
                      scale = recid.aft.w$scale,
                      distribution = recid.aft.w$dist)

recid$new_time <- new_time
recid$diff <- recid$new_time - recid$week

head(data.frame(recid$week, recid$new_time, recid$diff))
```

```
##    recid.week recid.new_time recid.diff
## 1          20       25.66776   5.667764
## 2          17       21.81760   4.817600
## 3          25       32.08471   7.084706
## 4          52       66.73619  14.736188
## 5          52       66.73619  14.736188
## 6          52       66.73619  14.736188
```

?