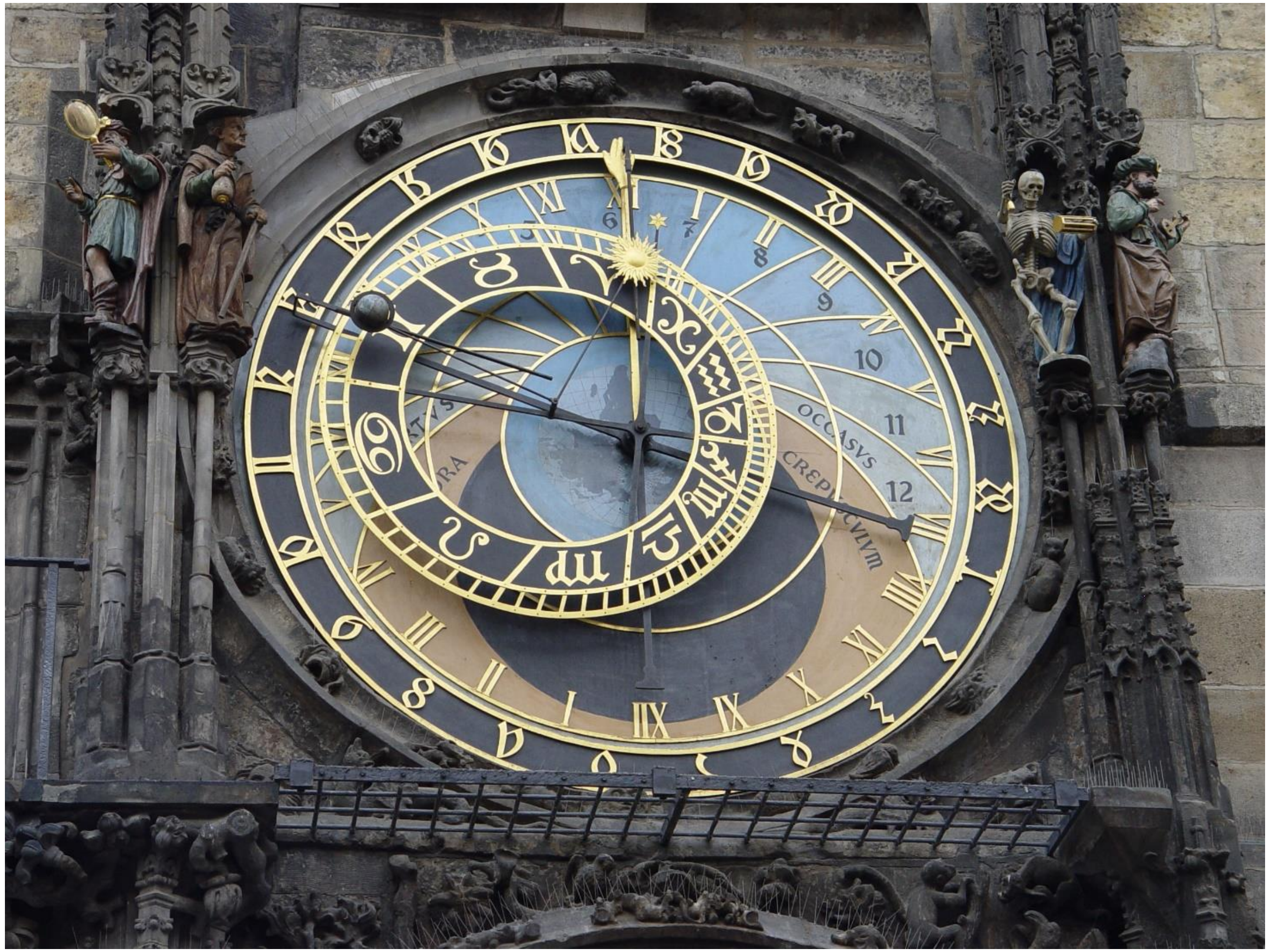
ACCELERATED FAILURE TIME MODEL (AFT)

"The bad news is time flies. The good news is you're the pilot." — **Michael Altshuler**



MODEL STRUCTURE

Parametric Models

- AFT models are **parametric** – we assume failure time (T) has a particular structure and distribution…we will be modeling time until failure (T)
- Kaplan-Meier estimation is **nonparametric** – makes no assumption on failure time.

- Parametric methods allow for more detailed/precise estimation than nonparametric methods **IF** the distribution is specified correctly.
  - Ex: Easier to estimate medians, survival & hazard functions.

Accelerated Failure Time Model

- The accelerated failure time (AFT) model is a regression that relates covariates (independent variables) to the event time *T*.
- The AFT model is a parametric model – depends on knowledge of the underlying distribution of the data.

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

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:

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

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:

$$\log T_i = \beta_0 + \underbrace{\beta_1 x_{i,1} + \cdots + \beta_k x_{i,k}}_{\text{Variables used to predict } T} + \sigma e_i$$

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:

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

Scale parameter of the distribution

Scale vs. Rate

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

Scale parameter for distribution

- Another common form is the multiplicative inverse of the scale, called the **rate**: $(1/\sigma)$.
- If σ is small, then events are not spread out → events happening close to one another or rate is large
- If σ is large, then events are spread out → smaller rate of events.

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:

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

Errors in the model

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
  - Distribution belongs to the location, scale family
  - Independence across observations

Common distributions for AFT model

Need a distribution that only takes on positive values and is right skewed (survival time distributions):

Weibull distribution
Lognormal distribution
Log-Logistic
Exponential

More to come on distributions later….

AFT Model – R with Lognormal

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

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
    data = recid, dist = "lognormal")
              Value     Std. Error    z      p
(Intercept)  4.0146     0.3897     10.30   < 2e-16
fin           0.3319     0.1657      2.00    0.04524
age           0.0333     0.0153      2.18    0.02959
mar           0.5609     0.2928      1.92    0.05541
prio         -0.0743     0.0264     -2.82    0.00481
Log(scale)   0.2723     0.0765      3.56    0.00037

Scale= 1.31


Log Normal distribution
Loglik(model)= -685.5          Loglik(intercept only)= -697.9
        Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
   data = recid, dist = "lognormal")
                Value    Std. Error    z      p
(Intercept)  4.0146    0.3897    10.30    < 2e-16
fin           0.3319    0.1657     2.00    0.04524
age           0.0333    0.0153     2.18    0.02959
mar           0.5609    0.2928     1.92    0.05541
prio         -0.0743    0.0264    -2.82    0.00481
Log(scale)   0.2723    0.0765     3.56    0.00037

Scale= 1.31

Log Normal distribution
Loglik(model)= -685.5           Loglik(intercept only)= -697.9
        Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

Parameter estimates

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
   data = recid, dist = "lognormal")
              Value     Std. Error   z      p
(Intercept)  4.0146     0.3897    10.30   < 2e-16
fin           0.3319     0.1657     2.00   0.04524
age           0.0333     0.0153     2.18   0.02959
mar           0.5609     0.2928     1.92   0.05541
prio         -0.0743     0.0264    -2.82   0.00481
Log(scale)   0.2723     0.0765     3.56   0.00037

Scale= 1.31

Log Normal distribution
Loglik(model)= -685.5          Loglik(intercept only)= -697.9
        Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

P-values for estimates in the model

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
   data = recid, dist = "lognormal")
              Value     Std. Error    z       p
(Intercept)  4.0146     0.3897     10.30   < 2e-16
fin           0.3319     0.1657      2.00    0.04524
age           0.0333     0.0153      2.18    0.02959
mar           0.5609     0.2928      1.92    0.05541
prio         -0.0743     0.0264     -2.82    0.00481
Log(scale)   0.2723     0.0765      3.56    0.00037

Scale= 1.31

Log Normal distribution
Loglik(model)= -685.5         Loglik(intercept only)= -697.9
      Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

Test to see if the “scale” (variance-like parameter) is equal to 1 (or log(1)=0)

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
   data = recid, dist = "lognormal")
              Value     Std. Error   z      p
(Intercept)  4.0146     0.3897    10.30   < 2e-16
fin           0.3319     0.1657     2.00    0.04524
age          0.0333     0.0153     2.18    0.02959
mar          0.5609     0.2928     1.92    0.05541
prio        -0.0743     0.0264    -2.82    0.00481
Log(scale)  0.2723     0.0765     3.56    0.00037

Scale= 1.31

Log Normal distribution
Loglik(model)= -685.5         Loglik(intercept only)= -697.9
       Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

Distribution assumed for the data

AFT Model - R

```
Call:
survreg(formula = Surv(week, arrest) ~ fin + age + mar + prio,
   data = recid, dist = "lognormal")
               Value     Std. Error    z       p
(Intercept)  4.0146     0.3897     10.30    < 2e-16
fin           0.3319     0.1657      2.00     0.04524
age           0.0333     0.0153      2.18     0.02959
mar           0.5609     0.2928      1.92     0.05541
prio         -0.0743     0.0264     -2.82     0.00481
Log(scale)   0.2723     0.0765      3.56     0.00037

Scale= 1.31

Log Normal distribution
Loglik(model)= -685.5          Loglik(intercept only)= -697.9
        Chisq= 24.85 on 4 degrees of freedom, p= 5.4e-05
Number of Newton-Raphson Iterations: 4
n= 432
```

Similar to an overall F test: H_0: Intercept model is appropriate (i.e. NO covariates are significant) versus H_A: At least one covariate is significant

INTERPRETATION

AFT Model Parameter Interpretation

- If a parameter estimate is **positive**, increases in that variable **increase** the expected survival time.
- If a parameter estimate is **negative**, increases in that variable **decrease** expected survival times.
- If a parameter estimate is **zero**, increases in that variable have **no impact** on expected survival times.

- $100 \times \left(e^{\beta} - 1\right)$ is the % increase in the expected survival time for each one-unit increase in the variable.

Recidivism Parameter Interpretation

Variable	β Estimate	$100\left(e^{\beta}-1\right)$
Financial Aid	0.3319	39.36%
Age at Release	0.0333	3.39%
Marital Status	0.5609	75.22%
Prior Convictions	-0.0743	-7.16%

Recidivism Parameter Interpretation

Variable	β Estimate	$100\left(e^{\beta} - 1\right)$
Financial Aid	0.3319	39.36%
Age at Release	0.0333	3.39%
Marital Status	0.5609	75.22%
Prior Convictions	-0.0743	-7.16%

For individuals who received financial aid, we expect their length of time until recidivism to be 39.4% longer than those who did not receive financial aid.

Recidivism Parameter Interpretation

Variable	β Estimate	$100(e^{\beta} - 1)$
Financial Aid	0.3319	39.36%
Age at Release	0.0333	$\boxed{3.39\%}$
Marital Status	0.5609	75.22%
Prior Convictions	-0.0743	-7.16%

For every increase in age by one year at time of release, we expect their length of time until recidivism to increase by 3.4%.

Recidivism Parameter Interpretation

Variable	β Estimate	$100\left(e^{\beta}-1\right)$
Financial Aid	0.3319	39.36%
Age at Release	0.0333	3.39%
Marital Status	0.5609	75.22%
Prior Convictions	-0.0743	-7.16%

For every increase in prior convictions, we expect their length of time until recidivism to decrease by 7.2%.

DISTRIBUTIONS

Common Distributions

Exponential Distribution

- Simplest distribution is the **exponential distribution** – constant hazard that doesn’t depend on time.
- Survival function: $S(t) = e^{-\lambda t}$
- Hazard function: $h(t) = \lambda$

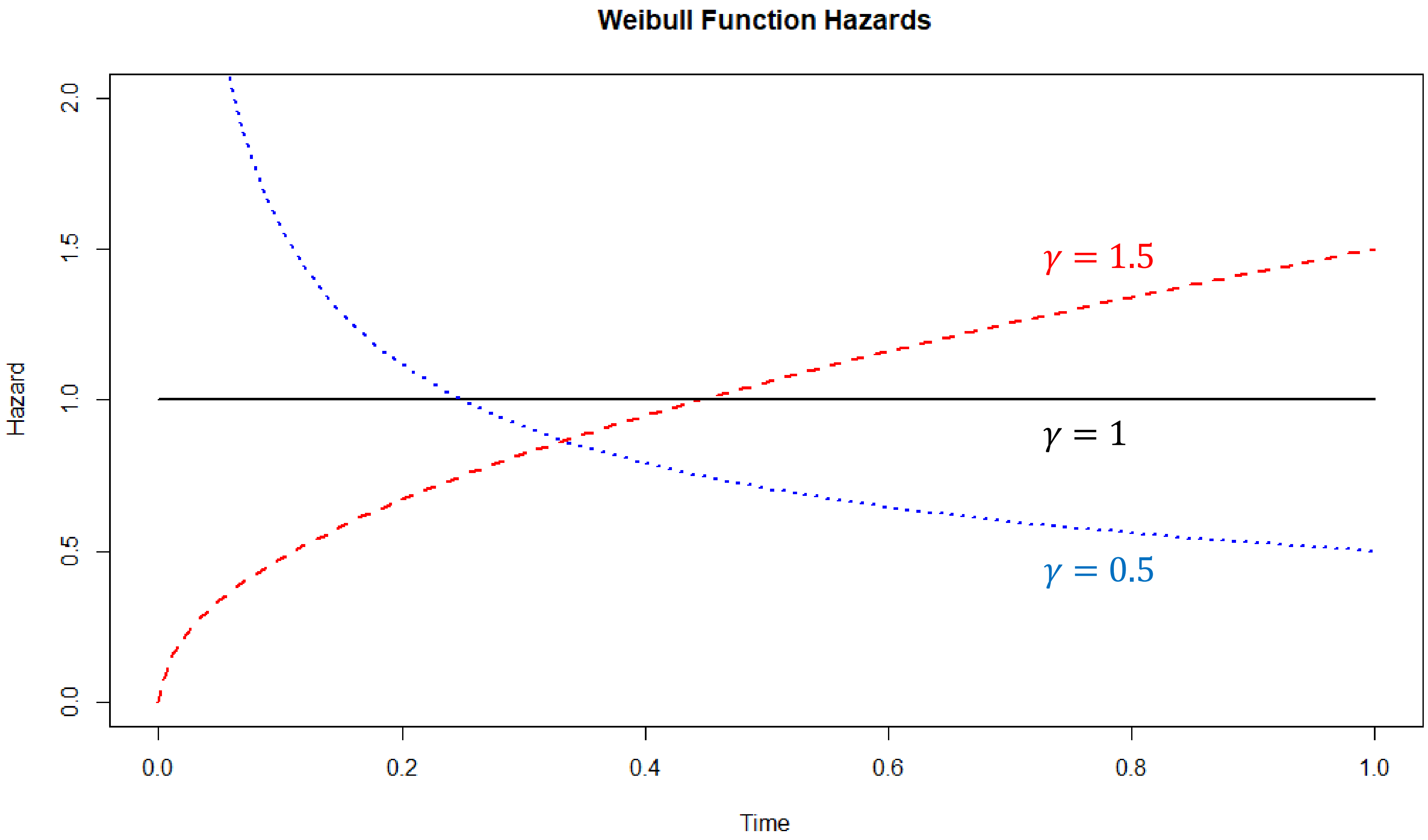
- Constant hazard commonly used when failures are completely random:
  - Light bulbs
  - Electronics
  - Etc.

Weibull Distribution

- Most commonly used distribution is the **Weibull** distribution, which has an additional parameter γ.
- Survival function: $S(t) = e^{-(\lambda t)^{\gamma}}$
- Hazard function: $h(t) = \lambda\gamma(\lambda t)^{\gamma-1}$

- The parameter γ is a rate parameter $(1/\sigma) > 0$ and determines whether the hazard increases or decreases with time:
  - $\gamma > 1$: hazard **increasing** with time (Ex: aging parts “wear out”)
  - $\gamma < 1$: hazard **decreasing** with time (Ex: post-surgery complications)

Weibull Distribution Hazards



Exponential vs. Weibull

- Hazard for Weibull is constant when $\gamma = 1$.
- Weibull distribution **IS** the exponential distribution when $\gamma = 1$!

- R: Log(scale) p-value → testing if $H_0 \, : \, \log(\gamma) = 0$

Weibull Distribution - R

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

R output

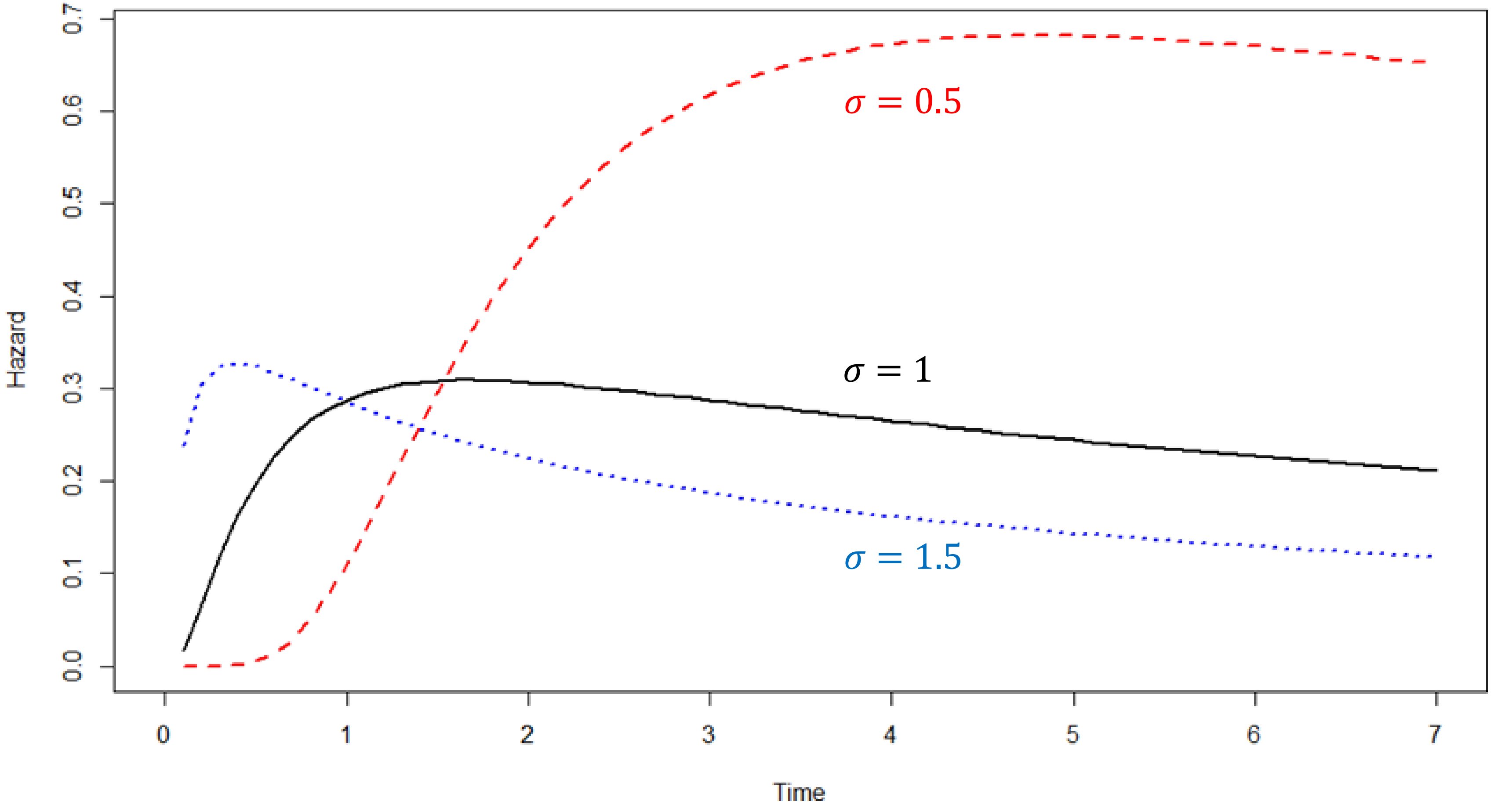
H_0: $\gamma = 1$ (or $\text{Log}(\gamma) = 0$)
H_A: $\gamma \neq 1$ (or $\text{Log}(\gamma) \neq 0$)

```
               Value     Std. Error      z          p
(Intercept)   3.8086    0.3762        10.12       < 2e-16
fin           0.2625    0.1376         1.91       0.05650
age           0.0400    0.0159         2.51       0.01210
wexp          0.1115    0.1516         0.74       0.46196
mar           0.3389     0.2725        1.24       0.21366
paro          0.0538     0.1394        0.39       0.69956
prio         -0.0646    0.0210        -3.08       0.00208
Log(scale)  -0.3383    0.0891        -3.80       0.00015

Scale= 0.713

Weibull distribution
Loglik(model)= -680.5   Loglik(intercept only)= -696.6
      Chisq= 32.28 on 6 degrees of freedom, p= 1.4e-05
Number of Newton-Raphson Iterations: 6
n= 432
```

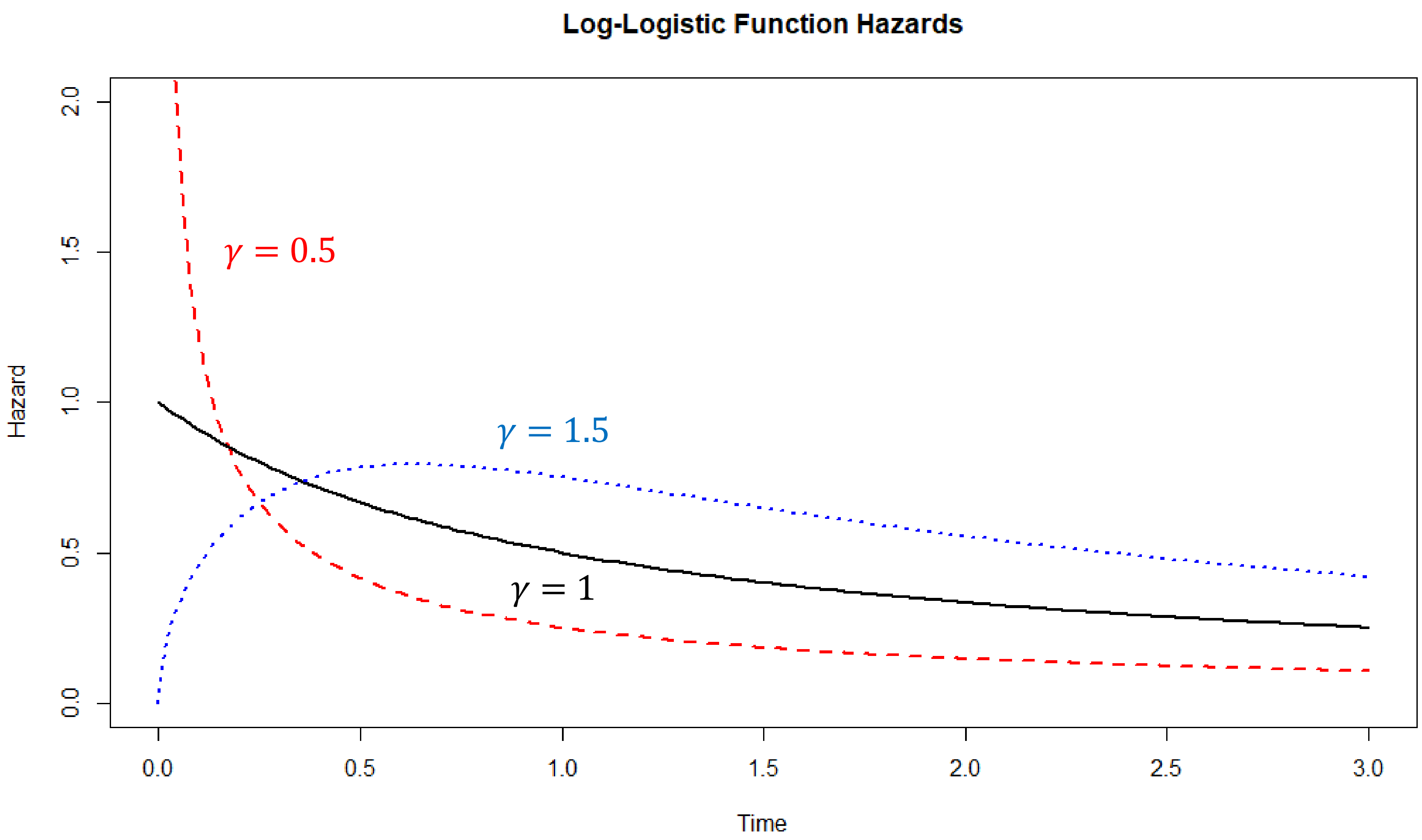
Other Distributions

- **Log-Normal Distribution:** If T has a log-normal distribution, then log(T) follows a normal distribution.
  - IF NO CENSORING, log-normal AFT = linear regression with $y = \log\ T$ are equivalent.
  - Hazard has different forms depending on the scale parameter
- **Log-Logistic Distribution:**
  - Hazard takes on different forms depending on the scale parameter
- **Generalized Gamma Distribution:**
  - Hazard takes on different forms depending on the scale parameter

Log-Normal Hazard

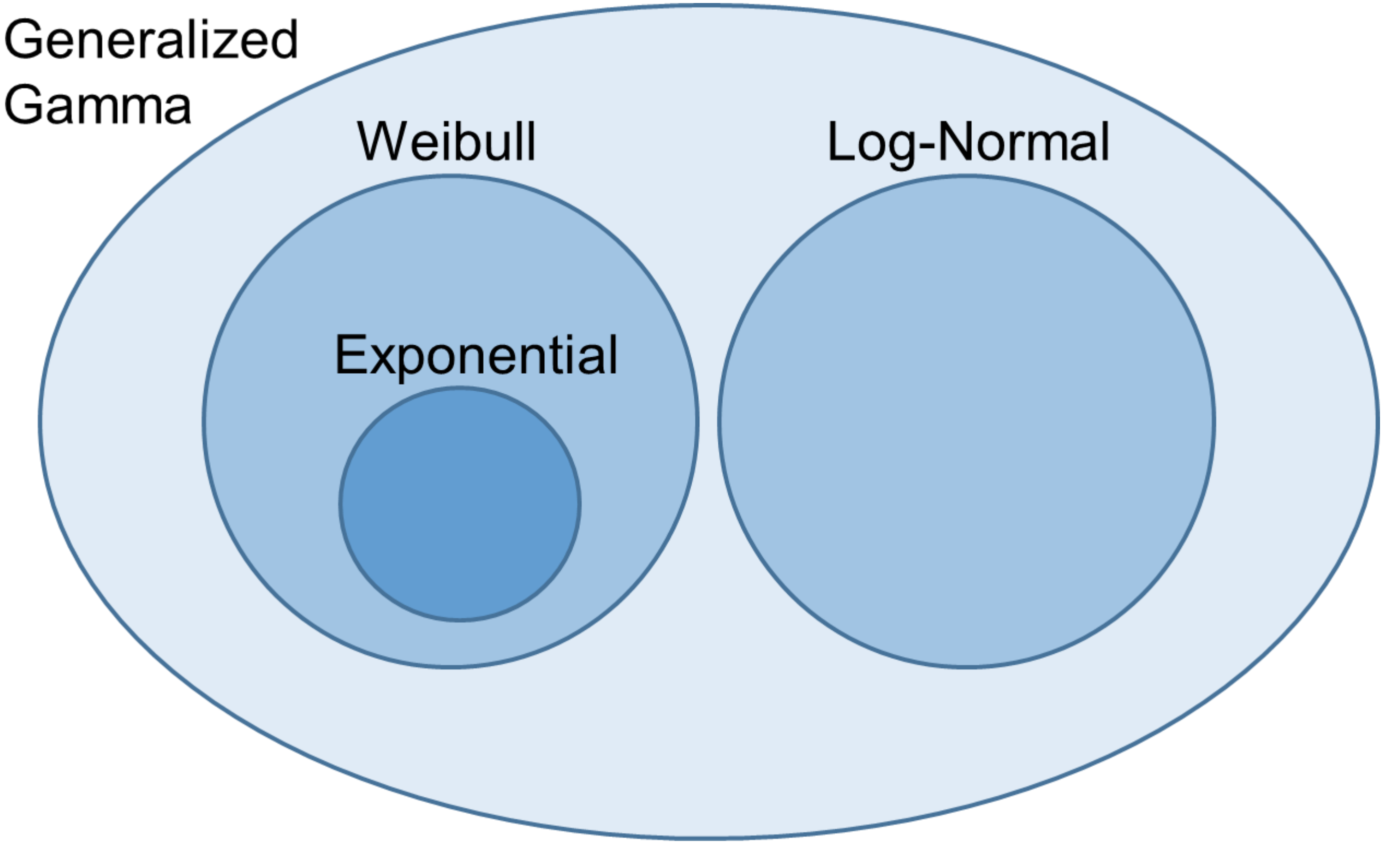


Log-Logistic Hazard



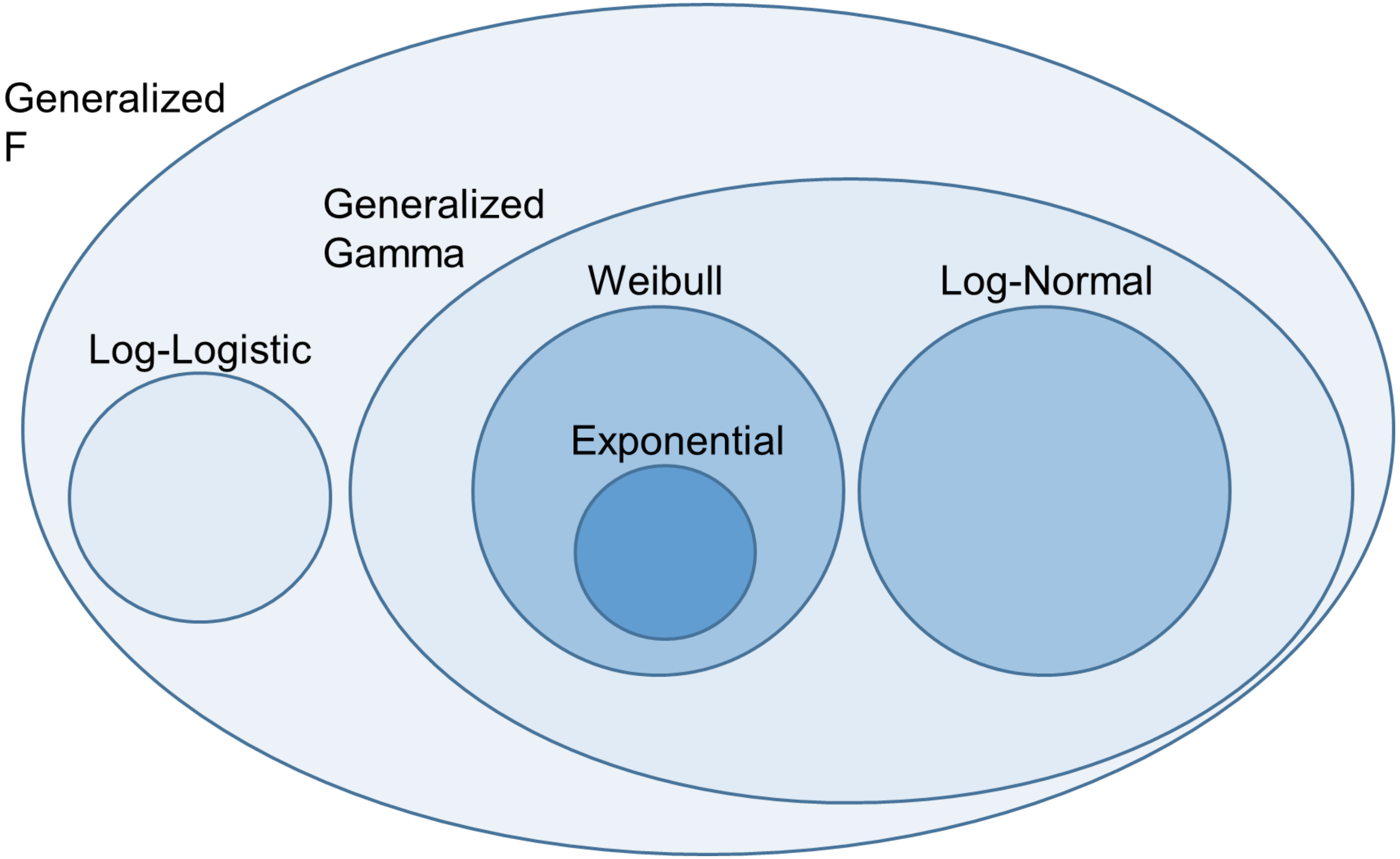
Other Distributions

- **Generalized Gamma Distribution:** Includes log-normal and Weibull as special cases.



Other Distributions

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



Distributions

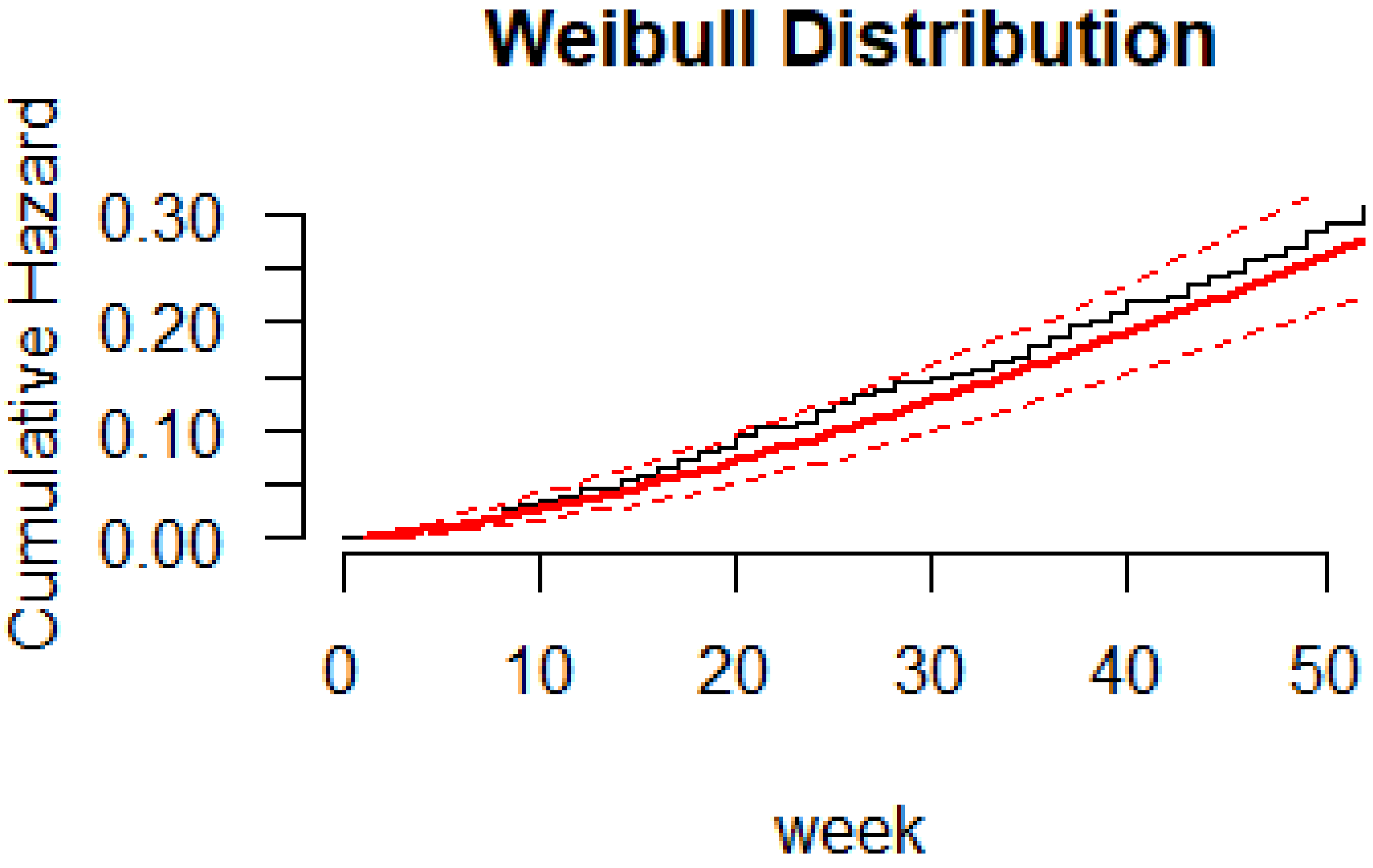
- Distributions are commonly checked two ways:
    1. Graphically (need to use the flexsurvreg and will compare cumulative hazard functions to actual data)
    2. Statistical Tests (if distributions are nested)

- There is **no guarantee** that your data will adequately match just one of the distributions here, or even any of them at all.
- Typically, use **full model** (all variables) since we don’t know which p-values are correct

Checking distributions - R

```
recid.aft.w <- flexsurvreg(Surv(week, arrest) ~ fin + age + 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")
```

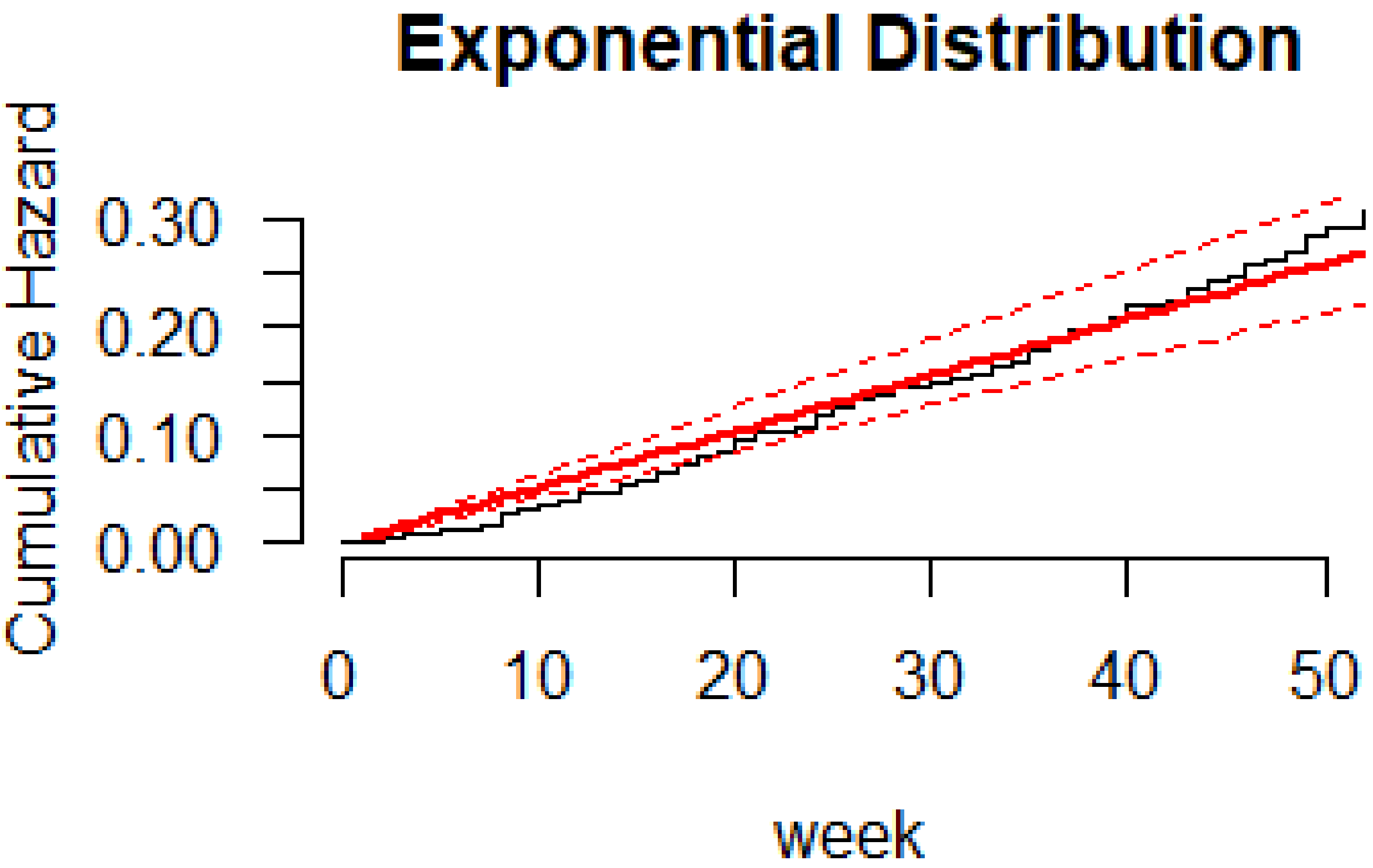
Cumulative Hazard - Weibull



Exponential

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

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

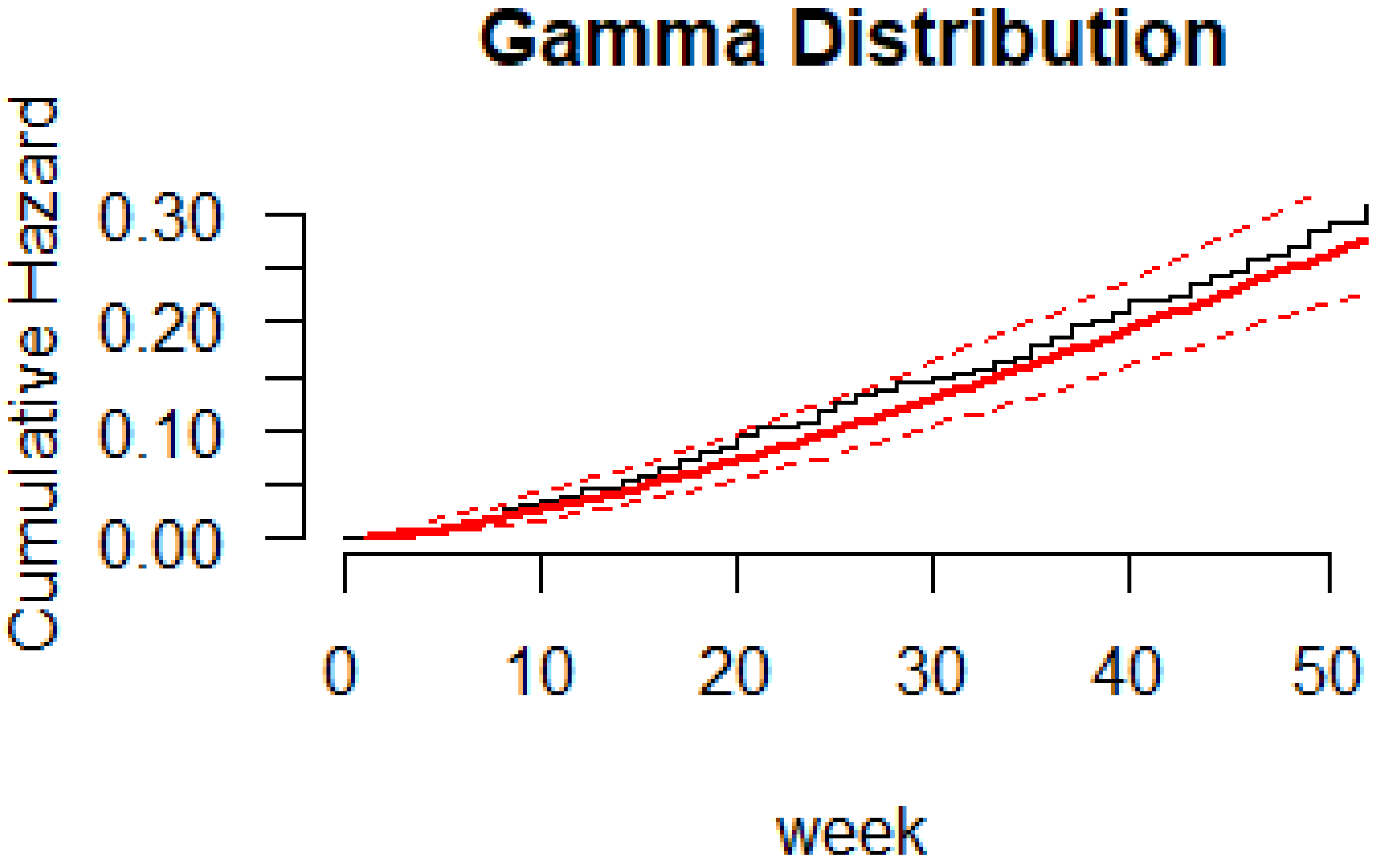
Cumulative Hazard - Exponential



Gamma

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

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

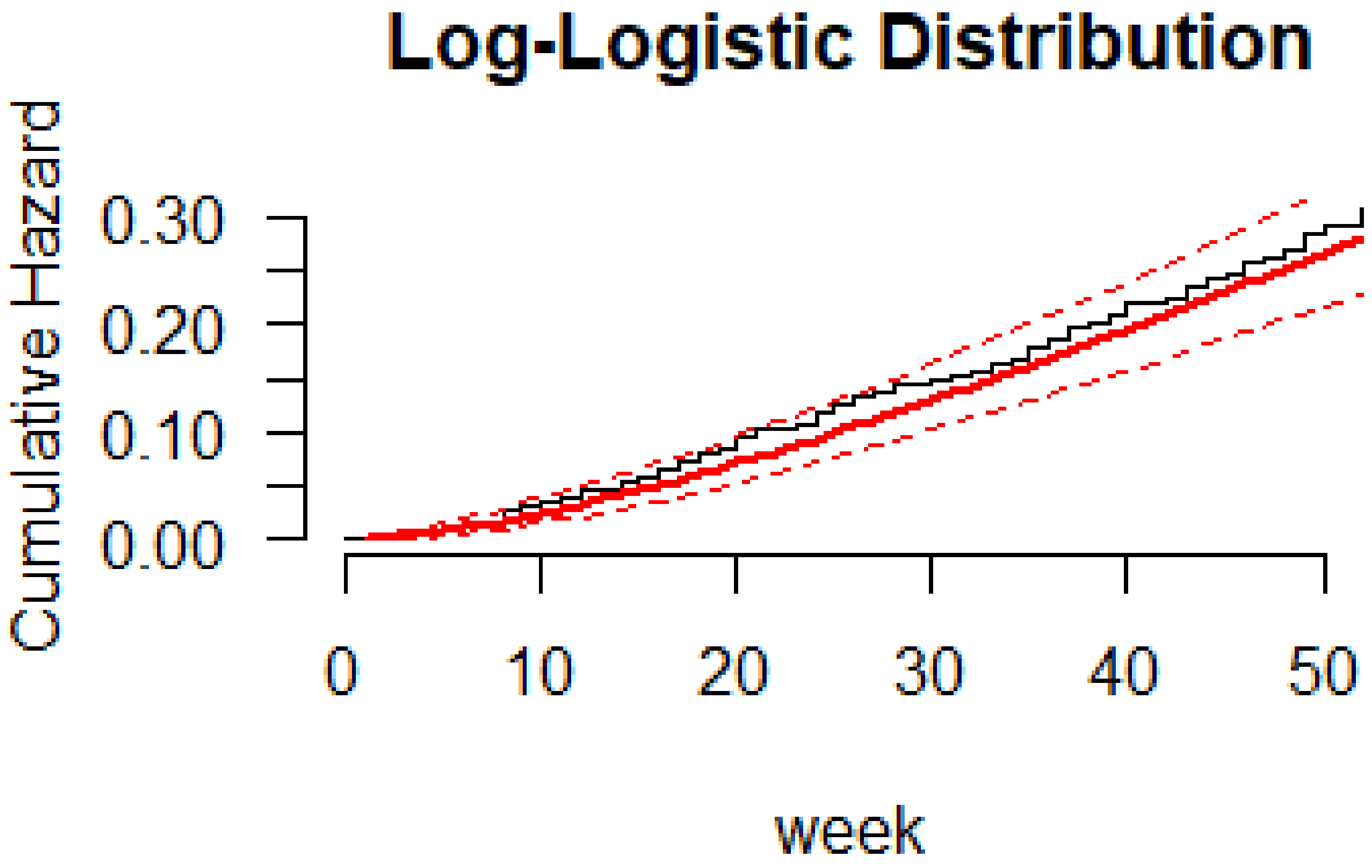
Cumulative Hazard - Gamma



Log-Logistic

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

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

Cumulative Hazard – Log Logistic

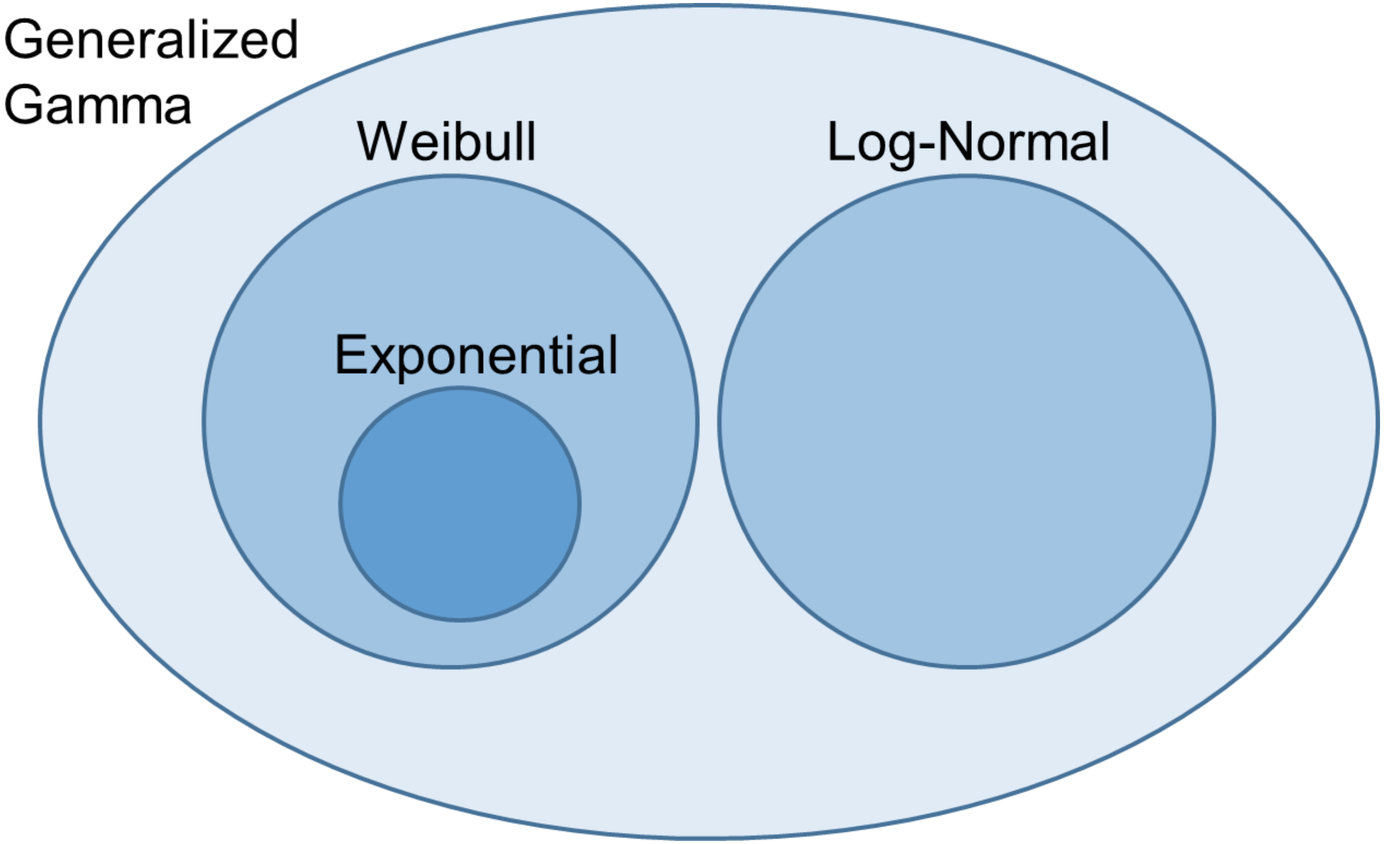


Distributions

- Distributions are commonly checked two ways:
  1. Graphically
  2. Statistical Tests…. For nested distributions
- There is **no guarantee** that your data will adequately match just one of the distributions here, or even any of them at all.

Nested Distributions!

- **Generalized Gamma Distribution:** Includes log-normal and Weibull as special cases.



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.

Code to get likelihoods…

```
like.e <- flexsurvreg(Surv(week, arrest) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"exp")$loglik

like.w <- flexsurvreg(Surv(week, arrest) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"weibull")$loglik

like.ln <- flexsurvreg(Surv(week, arrest ) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"lnorm")$loglik

like.g <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"gamma")$loglik

like.ll <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"llogis")$loglik
######DO NOT RUN the F –distribution….does NOT converge!!
like.f <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"genf")$loglik
```

Code to get likelihoods…

```
like.e <- flexsurvreg(Surv(week, arrest) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"exp")$loglik

like.w <- flexsurvreg(Surv(week, arrest) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"weibull")$loglik

like.ln <- flexsurvreg(Surv(week, arrest ) ~ fin + age +  wexp + mar + paro + prio, data = recid, dist =
"lnorm")$loglik

like.g <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"gamma")$loglik

like.ll <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"llogis")$loglik
######DO NOT RUN the F –distribution….does NOT converge!!
like.f <- flexsurvreg(Surv(week, arrest) ~ fin + age  + wexp + mar + paro + prio, data = recid, dist =
"genf")$loglik
```

Calculate p-values

```
Tests = c('Exp vs. Gam', 'Wei vs. Gam', 'LogN vs. Gam')
P_values = c(pval.e.g, pval.w.g, pval.ln.g)
cbind(Tests, P_values)
```

H_0: No “significant difference” in fit between the two distributions (Simpler distribution is fine)

H_A: Significant difference in fit between the two distributions (More complex distribution is needed)

```
          Tests         P_values
"Exp vs. Gam"       0.0019
"Wei vs. Gam"          1
"LogN vs. Gam"    0.0077
```

After finding best distribution

After finding best distribution, we can select the best model (by hand…forward selection or backward elimination)
Using the Weibull distribution, we will use the following model moving forward:

	Value	Std. Error	z	p
(Intercept)	3.7738	0.3581	10.54	< 2e-16
fin	**0.2495**	**0.1372**	**1.82**	**0.06901**
age	**0.0478**	**0.0154**	**3.11**	**0.00189**
prio	**-0.0698**	**0.0201**	**-3.47**	**0.00051**
Log(scale)	-0.3367	0.0892	-3.77	0.00016

Scale= 0.714

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.

Example Predictions

- Median survival time:
  - Find *t* such that $\hat{S}_i(t) = \ 0.5$
- The time by which *q%* of people with the same values for predictor variables have the event:
  - Find *t* such that $\hat{S}_i(t) = 1 - q$
- 20 week predicted survival probability:
  - $\hat{S}_i(20)$

- **CAREFUL:** $\hat{S}_i(t)$ is entirely determined by the distribution used so estimates WON'T be the same across different distributions.

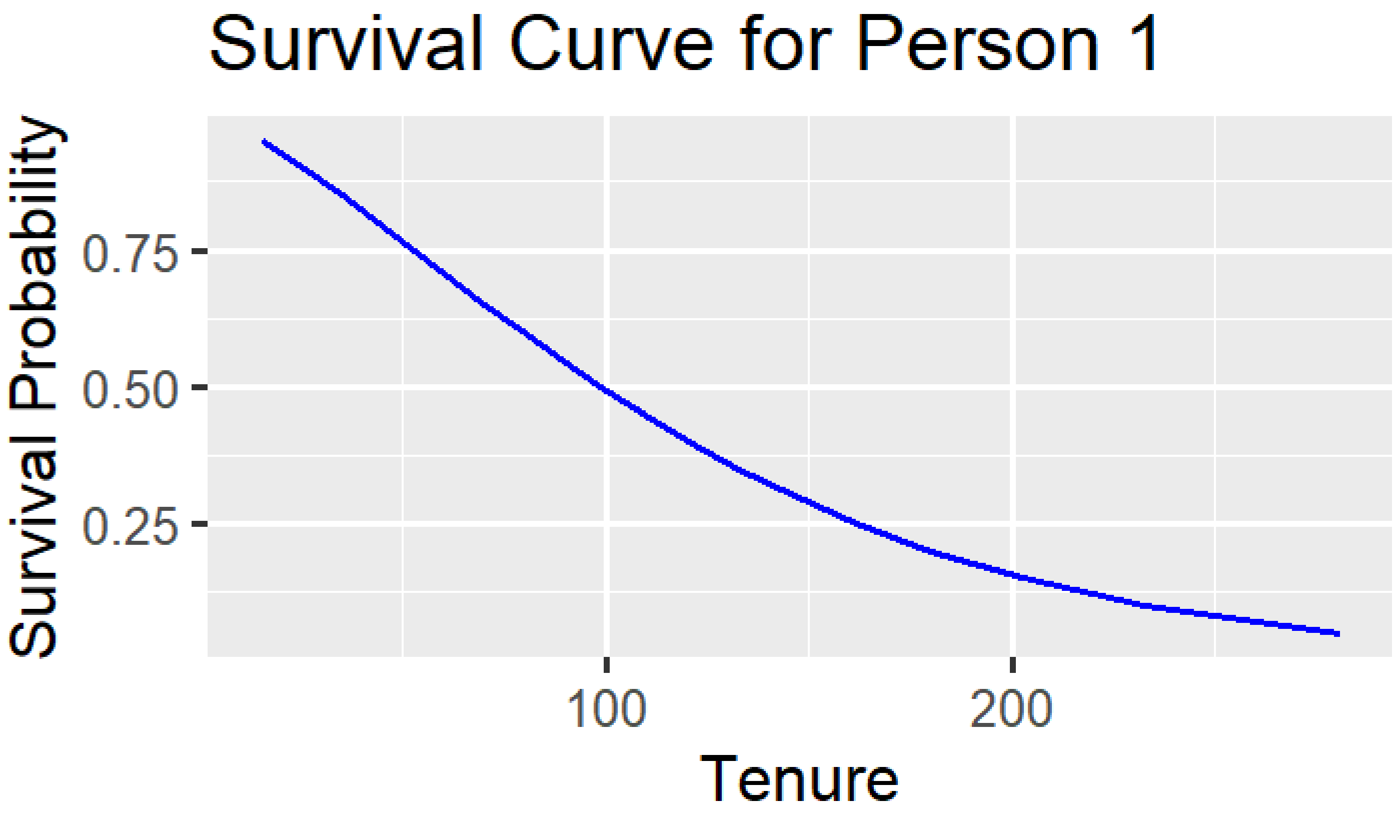
Predicting Survival Quantiles

```
# Predicted Survival Quantiles
recid.aft.w = survreg(Surv(week, arrest) ~ fin + age +prio, data = recid, dist = 'weibull')
summary(recid.aft.w)

#####Find the 25th, 50th and 75th percentile of survival curve for each individual
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
```

Survival Curve for First Individual

```
quant.prob=seq(0.05,0.95,by=0.05)
survprob = predict(recid.aft.w, type = "quantile",
se.fit = TRUE,p = quant.prob)
surv.prob=rev(quant.prob)

graph.dat=data.frame(cbind(survprob$fit[1,],surv.
prob))
colnames(graph.dat)=c("Tenure","SurvivalProb")

ggplot(graph.dat,aes(x=Tenure,y=SurvivalProb))+g
eom_line(color="blue")+labs(title="Survival Curve
for Person 1",x="Tenure",y="Survival Probability")
```



Predict mean survival time

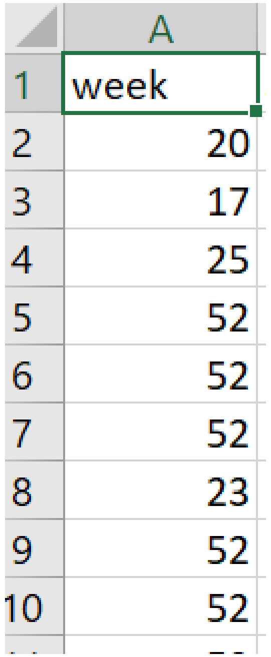
```
p.time.mean <- predict(recid.aft.w, type = "response", se.fit = TRUE)
head(p.time.mean$fit, n = 10)
```

```
[1] 128.26394   58.86229   43.55317   156.35349   87.52751
[6] 119.17415 143.73152  115.26040  81.92984   113.19494
```

Predict survival probabilities

```
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
[6] 0.7312118 0.9260438 0.7203354 0.5891529 0.7143008
```



	A
1	week
2	20
3	17
4	25
5	52
6	52
7	52
8	23
9	52
10	52

Predict survival probability at 10 weeks:

```
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
[6] 0.9693657
```

Predict Change in Event time

How much of an impact do we think there would be if we gave those individuals who did NOT have financial did have financial aid? We can use the tools that we just learned to explore this idea…

We will assume the “quantile” they fell on is still the same quantile, but we will use a new “linear predictor” (with financial aid =1)

```
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)
```

Predict Change in Event time

Now save new predict time to data set and find difference

Subset to only look at those that did NOT have financial to start!

```
recid$new_time = new_time
recid$diff = recid$new_time - recid$week

impact.fin=data.frame(recid$week, recid$new_time, recid$diff,recid$arrest,recid$fin)
colnames(impact.fin)=c("O.Week","N.Week","Diff","Arrest","Fin")
impact.fin2=subset(impact.fin,Arrest==1 & Fin==0)
head(impact.fin2)
```

Output

```
  O.Week    N.Week      Diff         Arrest  Fin
1       20    25.66776  5.667764      1    0
2       17    21.81760  4.817600      1    0
3       25    32.08471  7.084706      1    0
7       23    29.51793  6.517929      1    0
13     37     47.48536  10.485364   1    0
15     25     32.08471  7.084706      1    0
```