



# Our Survival Confidence Intervals are not the Same!

---

David Franklin  
Quintiles, Inc.  
Cambridge, MA

# Disclaimer

Any contents in this presentation does not reflect any policy, decisions, actions or guidance and are comments solely by the author.

# Introduction

# Basic Calculations

- Kaplan-Meier, the Survival Estimate was calculated using the formula

$$S(t) = \prod_{t_i < t} \frac{(n_i - d_i)}{n_i}$$

- The Standard Error could be calculated using

$$SE(t) = S(t) \sqrt{(1 - S(t)) / d_i}$$

- but the more usual calculation is done using Greenwood's formula

$$SE(t) = S(t) \sqrt{\sum_{i=1}^t \left[ \frac{d_i}{n_i(n_i - d_i)} \right]}$$

# Plain or Linear Method

$$P_k \pm 1.96 \text{ SE}$$

Data: 45\*, 75, 77, 84, 87, 88, 115, 117, 120\*, 120\*  
(\* indicates censoring)

# Introduction

ID#	Time to Event	Num at Risk ( $n_i$ )	Obs Events ( $d_i$ )	$(n_i - d_i)/n_i$	Survival Proportion $S(t)$	$d_j / (n_j - n_{j-1})$ ( $e$ )	SUM( $e$ )	SE(t)	LCL	UCL
	0	10	0							
1	45	10								
2	75	9	1	0.8889	0.8889	0.0139	0.0139	0.1048	0.6836	1.0942
3	77	8	1	0.8750	0.7778	0.0179	0.0317	0.1386	0.5062	1.0494
4	84	7	1	0.8571	0.6667	0.0238	0.0556	0.1571	0.3587	0.9747
5	87	6	1	0.8333	0.5556	0.0333	0.0889	0.1656	0.2309	0.8802
6	88	5	1	0.8000	0.4444	0.0500	0.1389	0.1656	0.1198	0.7691
7	115	4	1	0.7500	0.3333	0.0833	0.2222	0.1571	0.0253	0.6413
8									-	
	117	3	1	0.6667	0.2222	0.1667	0.3889	0.1386	0.0494	0.4938
9&10	120*	2	0							



# Log-Log Method

$$v(t) = \log(-\log(S(t)))$$

$$SE(t) = \sqrt{\frac{\sum_{i=1}^t \left[ \frac{d_i}{n_i(n_i - d_i)} \right]}{\left[ \sum_{i=1}^t \log\left(\frac{n_i - d_i}{n_i}\right) \right]^2}}$$

The 95% CI is also calculated a little differently,

$$[S(t)]^{e^{+/-1.96SE}}$$

# Log-Log Method

ID#	Time to Event	Num at Risk ( $n_i$ )	Obs Events ( $d_i$ )	Surv Prop $S(t)$	SUM (e)	$\log(n_i - d_i)/n_i$ (l)	SUM(l)	SE*	LCL	UCL
	0	10	0							
1	45	10	0							
2	75	9	1	0.8889	0.0317	-0.1178	-0.1178	1.0006	0.4330	0.9836
3	77	8	1	0.7778	0.0556	-0.1335	-0.2513	0.7090	0.3647	0.9393
4	84	7	1	0.6667	0.0889	-0.1542	-0.4055	0.5813	0.2817	0.8783
5	87	6	1	0.5556	0.1389	-0.1823	-0.5878	0.5072	0.2042	0.8045
6	88	5	1	0.4444	0.2222	-0.2231	-0.8109	0.4596	0.1359	0.7193
7	115	4	1	0.3333	0.3889	-0.2877	-1.0986	0.4291	0.0783	0.6226
8	117	3	1	0.2222		-0.4055	-1.5041	0.4146	0.0337	0.5131
9&10	120*									

\* LOG(SE)



# SAS from 9.1x Onwards

- arcsine-square root (ASINSQRT), log-log (LOGLOG), linear (LINEAR), logarithmic (LOG) or logit (LOGIT) transformation
- SAS v9.1x specified option in the Survival Statement, from v9.2 an option called CONFTYPE=

Time	Censor	Survival		LINEAR		LOGLOG		LOG		LOGIT		ASINSQRT	
		Estimate	SE	LCL	UCL	LCL	UCL	LCL	UCL	LCL	UCL	LCL	UCL
0	.	1.00000	.	.	.	.	.	.	.	.	.	.	.
45	1	1.00000	.	.	.	.	.	.	.	.	.	.	.
75	0	0.8889	0.1048	0.6836	1.0942	0.4330	0.9836	0.7056	1.1199	0.1112	0.9846	0.6178	0.9998
77	0	0.7778	0.1386	0.5062	1.0494	0.3648	0.9393	0.5485	1.1029	0.1720	0.9440	0.4679	0.9733
84	0	0.6667	0.1571	0.3587	0.9746	0.2817	0.8783	0.4200	1.0581	0.2001	0.8889	0.3458	0.9189
87	0	0.5556	0.1656	0.2309	0.8802	0.2042	0.8045	0.3097	0.9966	0.2117	0.8232	0.2421	0.8461
88	0	0.4444	0.1656	0.1198	0.7691	0.1359	0.7193	0.2141	0.9227	0.2117	0.7487	0.1539	0.7579
115	0	0.3333	0.1571	0.0254	0.6413	0.0783	0.6226	0.1323	0.8397	0.2001	0.6666	0.0811	0.6542
117	0	0.2222	0.1386	-0.0494	0.4938	0.0337	0.5131	0.0655	0.7544	0.1720	0.5790	0.0267	0.5321
120	1	.	.	.	.	.	.	.	.	.	.	.	.
120	1	.	.	.	.	.	.	.	.	.	.	.	.

# Code Using Base SAS

```
retain _prestart &num survival . _linearB 0;
_alpha = probit(1-&alpha/2);
_entered_period=_prestart;
_event_period=_sact;
_censor_period=_sout-_sact;
if _event_period>0 then do;
  _survesta=(_entered_period-_event_period)/_entered_period;
  if missing(survival) then survival=_survesta;
  else survival=survival*_survesta;

  _LinearA=_event_period/(_entered_period*( _entered_period-_event_period));
  _LinearB=_LinearB+_LinearA;
  stderr=sqrt(survival*survival*_LinearB);
```

# Code Using Base SAS

\*Linear;

```
lcl_linear=survival-_alpha*stderr;  
ucl_linear=survival+_alpha*stderr;
```

\*LOG-LOG;

```
band_loglog = _alpha * sqrt(stderr*stderr / ((survival*log(survival))**2));  
lcl_loglog = survival ** (exp(band_loglog));  
ucl_loglog = survival ** (exp(-band_loglog));
```

\*LOG;

```
band_log = _alpha* sqrt(stderr*stderr / (survival**2));  
lcl_log = survival*exp(-band_log);  
ucl_log = survival*exp(band_log);
```

# Code Using Base SAS

```
*LOTIT;
lcl_logit = survival /
  (survival + (survival)*exp(_alpha*stderr / (survival * (1-survival))));
ucl_logit = survival /
  (survival + (1-survival)* exp(-_alpha*stderr / (survival * (1-survival))));

*ASINSQRT;
lcl_asinsqrt = (sin(max(0, arsin(sqrt(survival))-
  _alpha*stderr/(2*sqrt(survival*(1-survival)))))**2;
ucl_asinsqrt = (sin(min(constant('pi')/2, arsin(sqrt(survival)) +
  _alpha*stderr/(2*sqrt(survival*(1-survival)))))**2; . . . . .
```

# Conclusion

# Pharma

# SUG

SAN DIEGO

2014



David Franklin  
Quintiles, Inc.  
dfranklinnh@gmail.com