Approved for public release; distribution is unlimited.

INSTITUTE FOR DEFENSE ANALYSES

Censored Data Analysis Methods for Performance Data: A Tutorial

V. Bram Lillard

April 2016
Approved for public release.
IDA Document NS D-5811
Log: H 16-000561

INSTITUTE FOR DEFENSE ANALYSES
4850 Mark Center Drive
Alexandria, Virginia 22311-1882

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

The Institute for Defense Analyses is a non-profit corporation that operates three federally funded research and development centers to provide objective analyses of national security issues, particularly those requiring scientific and technical expertise, and conduct related research on other national challenges.

About This Publication

Binomial metrics like probability-to-detect or probability-to-hit typically provide operationally meaningful and easy to interpret test outcomes. However, they are information-poor metrics and extremely expensive to test. The standard power calculations to size a test employ hypothesis tests, which typically result in many tens to hundreds of runs. In addition to being expensive, the test is most likely inadequate for characterizing performance over a variety of conditions due to the inherently large statistical uncertainties associated with binomial metrics. A solution is to convert to a continuous variable, such as miss distance or time-to-detect. The common objection to switching to a continuous variable is that the hit/miss or detect/non-detect binomial information is lost, when the fraction of misses/no-detects is often the most important aspect of characterizing system performance. Furthermore, the new continuous metric appears to no longer be connected to the requirements document, which was stated in terms of a probability. These difficulties can be overcome with the use of censored data analysis. This presentation will illustrate the concepts and benefits of this approach, and will illustrate a simple analysis with data, including power calculations to show the cost savings for employing the methodology.

Copyright Notice

© 2016 Institute for Defense Analyses

4850 Mark Center Drive, Alexandria, Virginia 22311-1882 • (703) 845-2000.

This material may be reproduced by or for the U.S. Government pursuant to the copyright license under the clause at DFARS 252.227-7013 (a)(16) [Jun 2013].

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

I N S T I T U T E F O R D E F E N S E A N A L Y S E S

IDA Document NS D-5811

Censored Data Analysis Methods for Performance Data: A Tutorial

V. Bram Lillard

Approved for public release; distribution is unlimited.

Censored Data Analysis for Performance Data: A Tutorial

V. Bram Lillard
Institute for Defense Analyses

Approved for public release; distribution is unlimited.

IDA

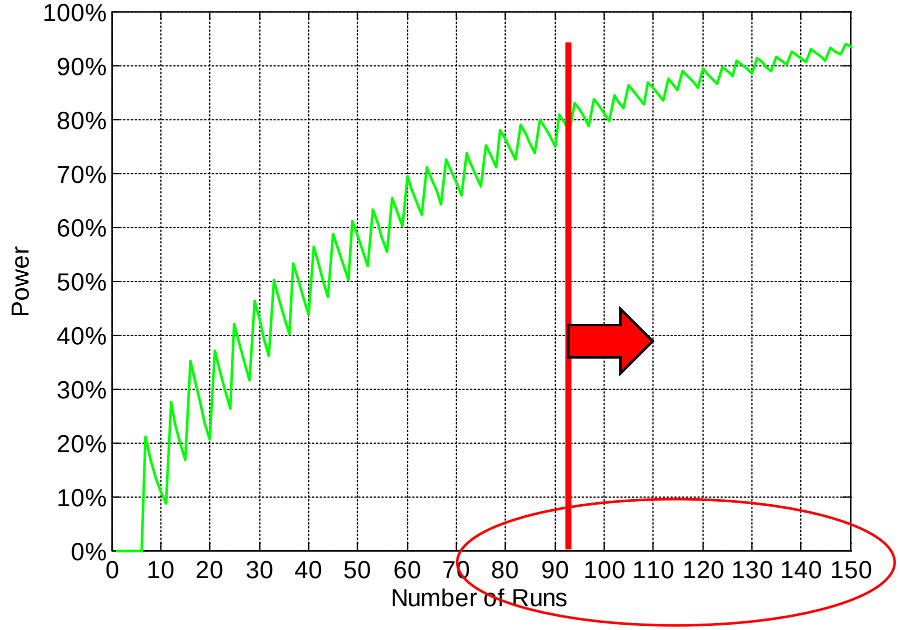
The Binomial Conundrum

- **Testing for a binary metric requires large sample sizes**

Sample Size Requirements

Sample Size	90% Confidence Interval Width (p = 0.5)	90% Confidence Interval Width (p = 0.8)
10	± 26%	± 21%
50	± 11.6%	± 9.3%
100	± 8.2%	± 6.6%
500	± 3.7%	± 2.9%

Power Calculation, 90% confidence, Performance 10% better than threshold



- **Difficult (impossible?) to achieve acceptable power for factor analysis unless many runs *(often >100)* can be resourced**
  - Non-starter for implementing DOE concepts (characterizing performance across multiple conditions)

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Continuous Metrics: An informative test solution

- **Chemical Agent Detector**
  - Requirement: Probability of detection greater than 85% within one minute
  - Original response metric: Detect/Non-detect
  - Replacement: Time until detection
- **Submarine Mine Detection**
  - Requirement: Probability of detection greater than 80% outside 200 meters
  - Original response metric: Detect/Non-detect
  - Replacement: Detection range
- **Missile System**
  - Requirement: Probability of hit at least 90%
  - Original response metric: Hit/Miss
  - Replacement: Missile miss distance

Surrogate continuous metrics provide much more information!

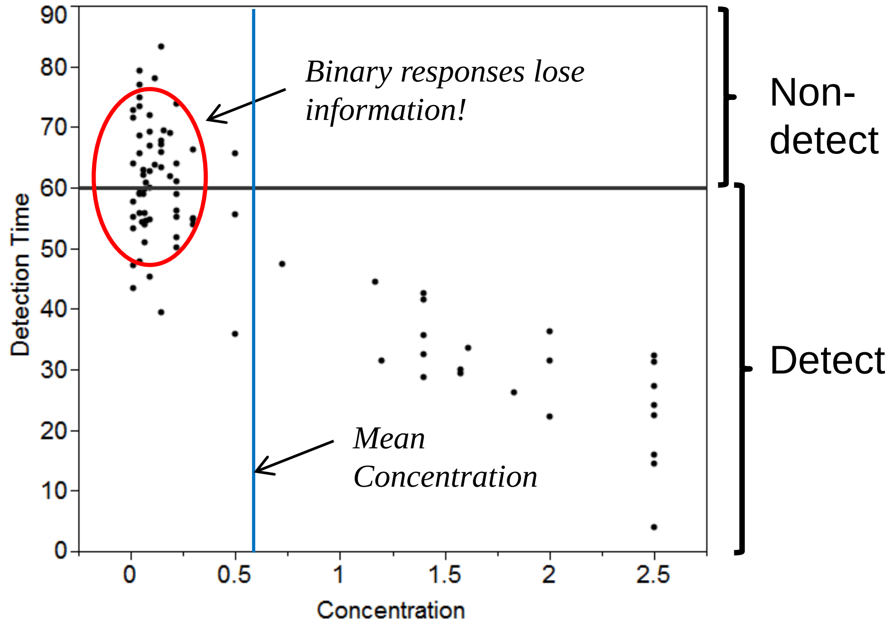
Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Chemical Agent Detector Results

- Estimate the probability of detection at 60 seconds at the mean concentration
- Detection times and detect/non-detect information recorded
- Binary analysis results in **300% increase** in confidence interval width

Data is for Illustration only



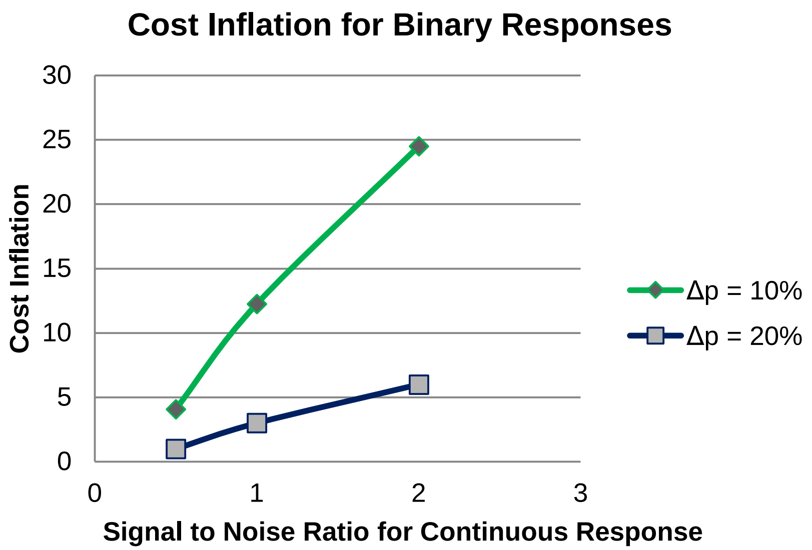
Response	Probability of Detection within 60 seconds at mean	Lower 90% Confidence Bound	Upper 90% Confidence Bound	Confidence Interval Width
Binary (Detect: Yes/No)	83.5%	60.5%	94.4%	33.9%
Continuous (Time)	91.0%	86.3%	94.5%	8.2%

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Solutions to the Binomial Conundrum

- **Recast Binomial metric (e.g., probability of detection) as a *continuous metric* (e.g., time-to-detect)**
  – Others: detection range, miss distance
- **Significant cost savings realized, plus the continuous metric provides useful information to the evaluator/users**



- **Challenges:**
  – How to handle ***non-detects***/misses?
    » Typical DOE methods (linear regression) require an actual measurement of the variable for every event
    » Can not force the test to get detection ranges, or throw out events – non-detects are important test results!
  – Common concern: Switching to the continuous measure seems to eliminate the ability to evaluate the requirement
    » E.g., we measured time-to-detect and calculated a mean, how do we determine if the system met it's KPP: P_{detect}>0.50?)

Approved for public release; distribution is unlimited.

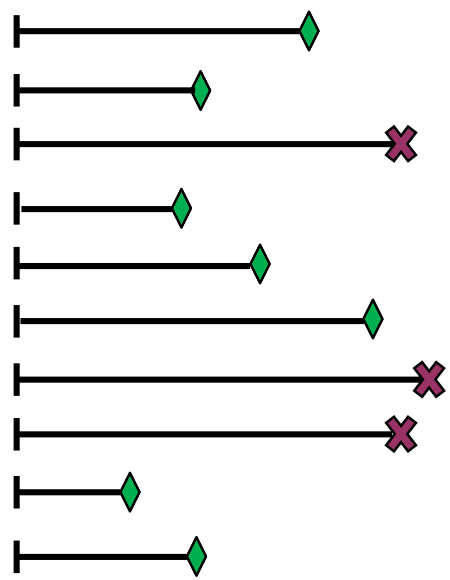
Approved for public release; distribution is unlimited.

Using Continuous Data (with non-detects)

- **Censored data = we didn't observe the detection directly, but we expect it will occur if the test had continued**
  - We cannot make an exact measurement, but there <u>is</u> information we can use!
  - Same concept as a time-terminated reliability trials (failure data)

Run No.	Result	Result Code
1	Detected Target	1
2	Detected Target	1
3	No detect	0
4	Detected Target	1
5	Detected Target	1
6	Detected Target	1
7	No detect	0
8	No detect	0
9	Detected Target	1
10	Detected Target	1

Corresponding Timelines



Run No.	Time of Detection (hours after COMEX)
1	**4.4**
2	**2.7**
3	**>6.1**
4	**2.5**
5	**3.5**
6	**5.3**
7	**>6.2**
8	**>5.8**
9	**1.8**
10	**2.7**

= Detect *= No-Detect*

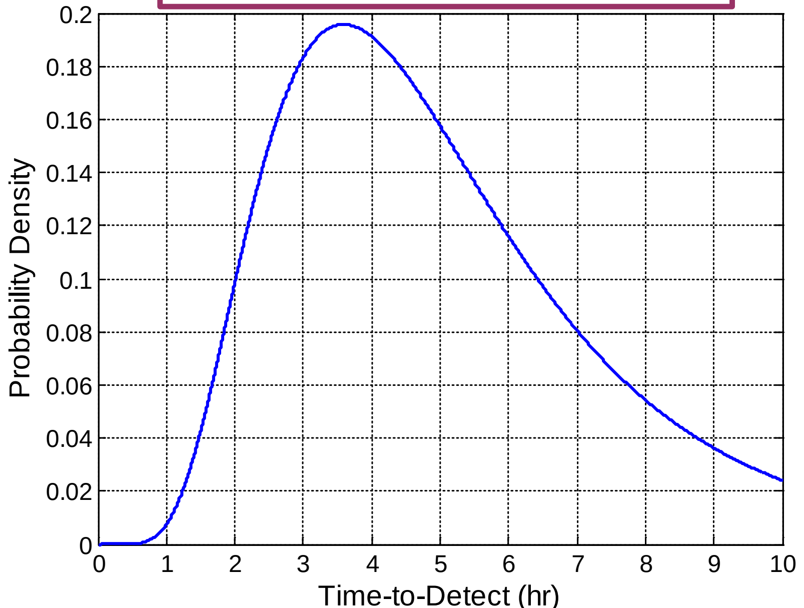
Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

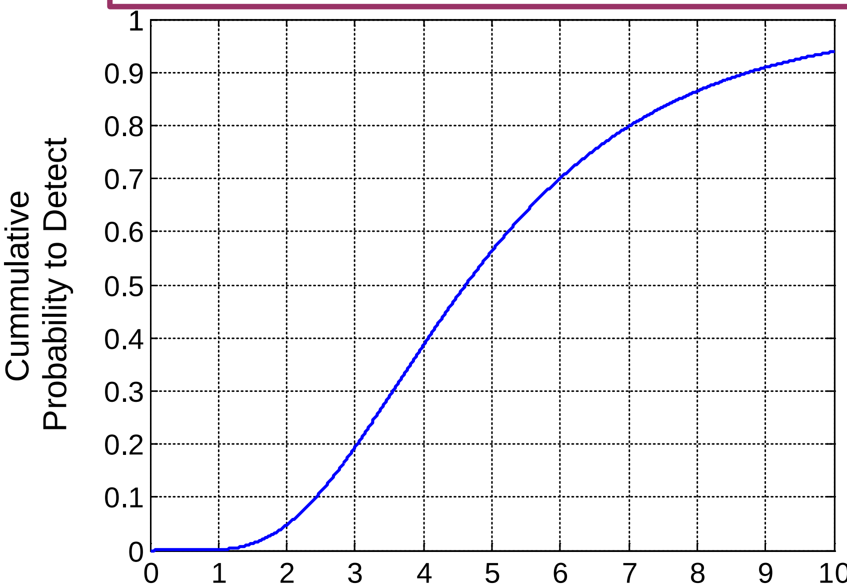
Parameterizing Data

- **Assume that the time data come from an underlying distribution, such as the log-normal distribution**
  - Other distributions may apply – *<u>must consider carefully</u>*, and check the assumption when data are analyzed
- **That parameterization will enable us to <u>link</u> the time metric to the probability of detection metric.**

Probability Density Function



Cumulative Distribution Function (CDF)



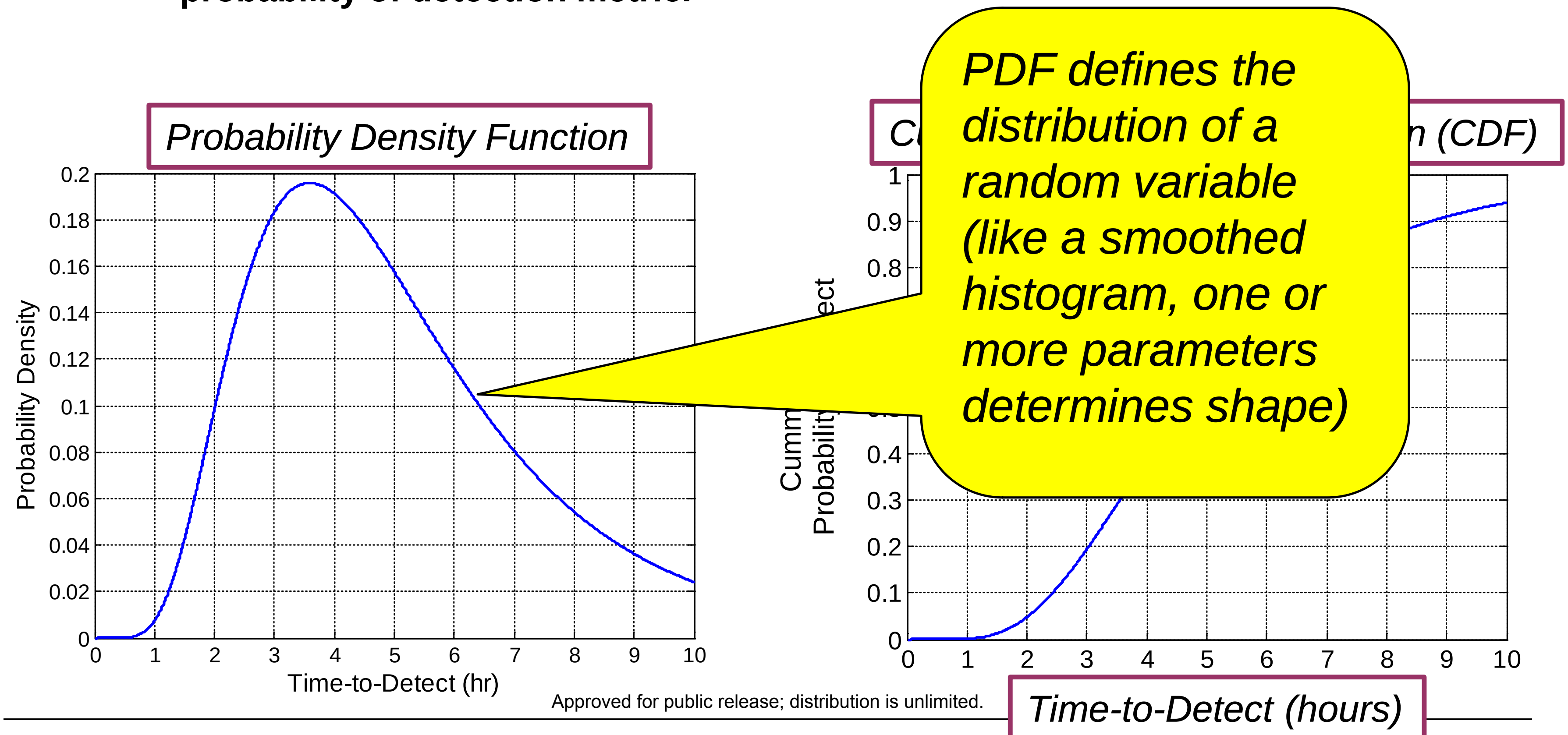
Time-to-Detect (hours)

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Parameterizing Data

- **Assume that the time data come from an underlying distribution, such as the log-normal distribution**
  - Other distributions may apply – *<u>must consider carefully</u>*, and check the assumption when data are analyzed
- **That parameterization will enable us to <u>link</u> the time metric to the probability of detection metric.**



Approved for public release; distribution is unlimited.

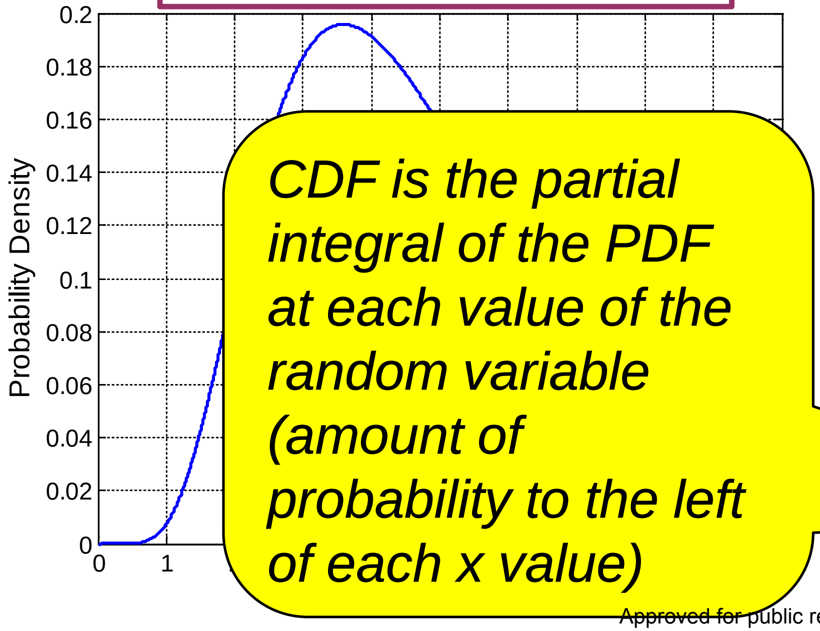
Approved for public release; distribution is unlimited.

Parameterizing Data

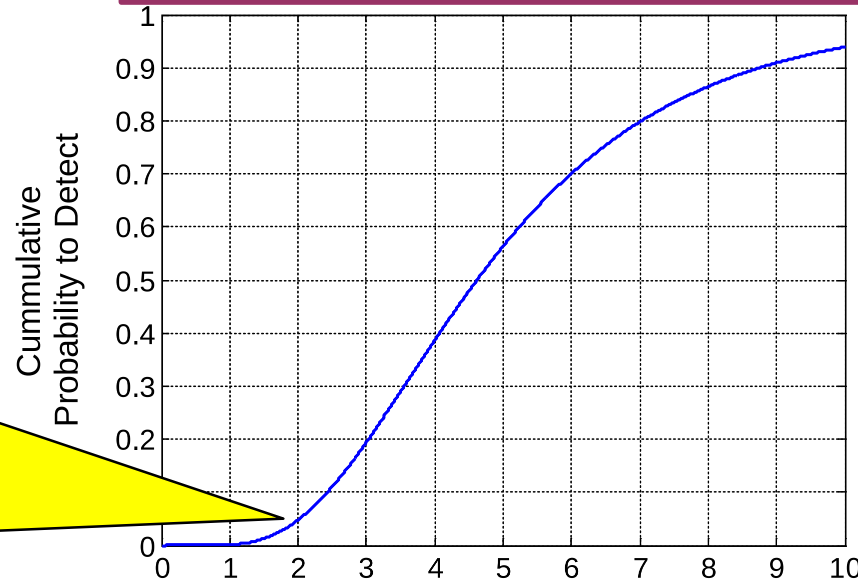
- **Assume that the time data come from an underlying distribution, such as the log-normal distribution**
  - Other distributions may apply – *<u>must consider carefully</u>*, and check the assumption when data are analyzed

- **That parameterization will enable us to <u>link</u> the time metric to the probability of detection metric.**

Probability Density Function



Cumulative Distribution Function (CDF)



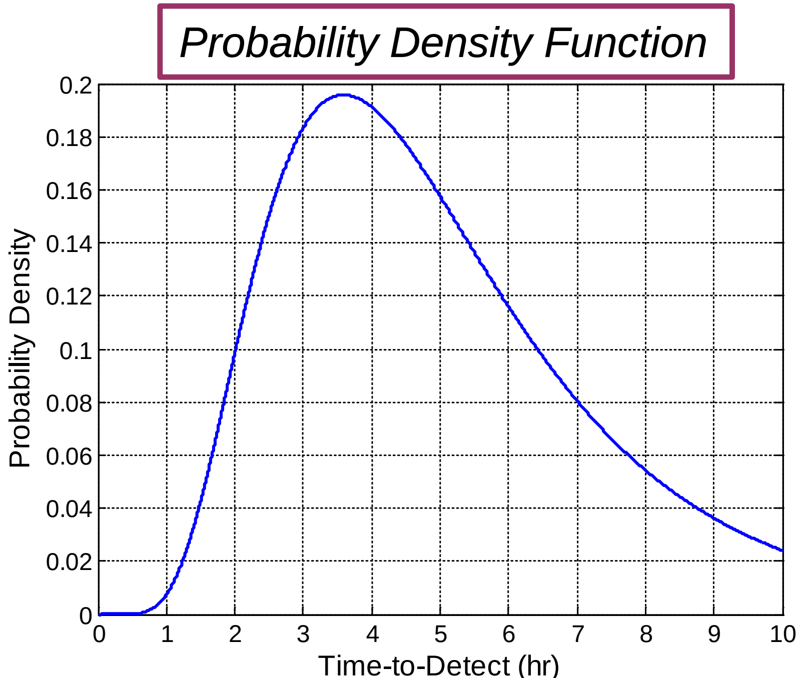
Time-to-Detect (hours)

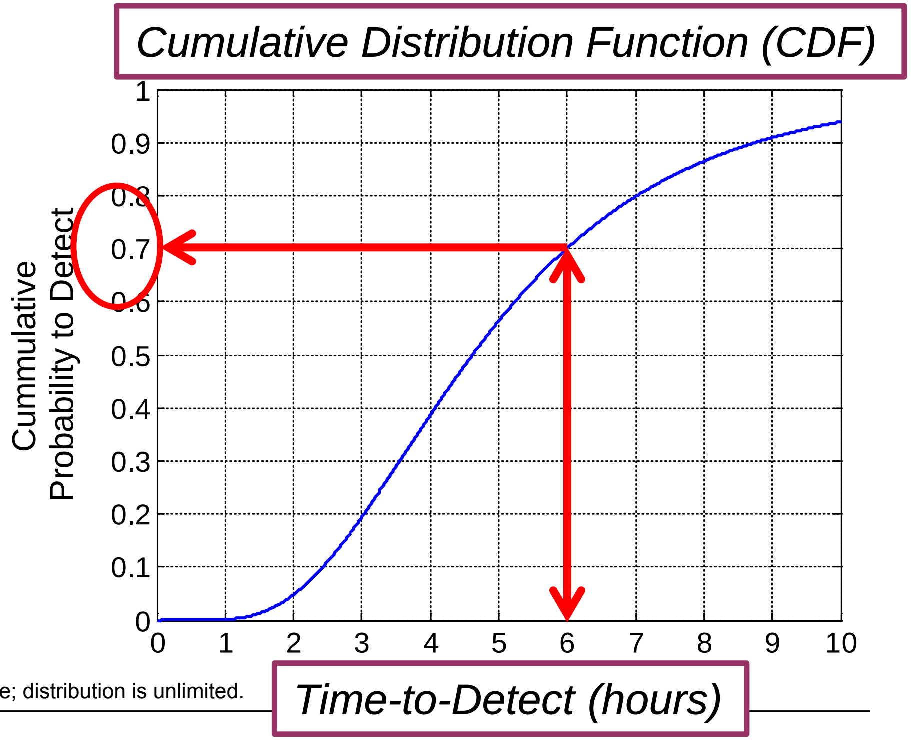
Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Parameterizing Data

- **Example: Aircraft must detect the target within it's nominal time on station (6-hours)**
  - Binomial metric was detect/non-detect within time-on-station
- **If we determine the shape of this curve (i.e., determine the parameters of the PDF/CDF), we can use the time metric to determine the probability to detect!**





Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

New Goal

- **Goal of our data analysis: determine the parameters of the distribution**
  - Once the CDF's shape is known, can determine:
    - » Median/Mean time to detect
    - » And… translate back to the binomial metric (probability to detect)

- **Most common and generalized technique for determining the parameters is via *maximum likelihood methodology***
  - A Likelihood is simply a function that defines how "likely" a particular value for a parameter is given the specific data we've observed

parameters TBD *data* *PDF*

$$L(\boldsymbol{\theta}|\boldsymbol{t}) = \prod_{i=1}^{\#\,data\,points} L(\boldsymbol{\theta}|t_i) = \prod_{i=1}^{\#\,data\,points} f(t_i|\boldsymbol{\theta})$$

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

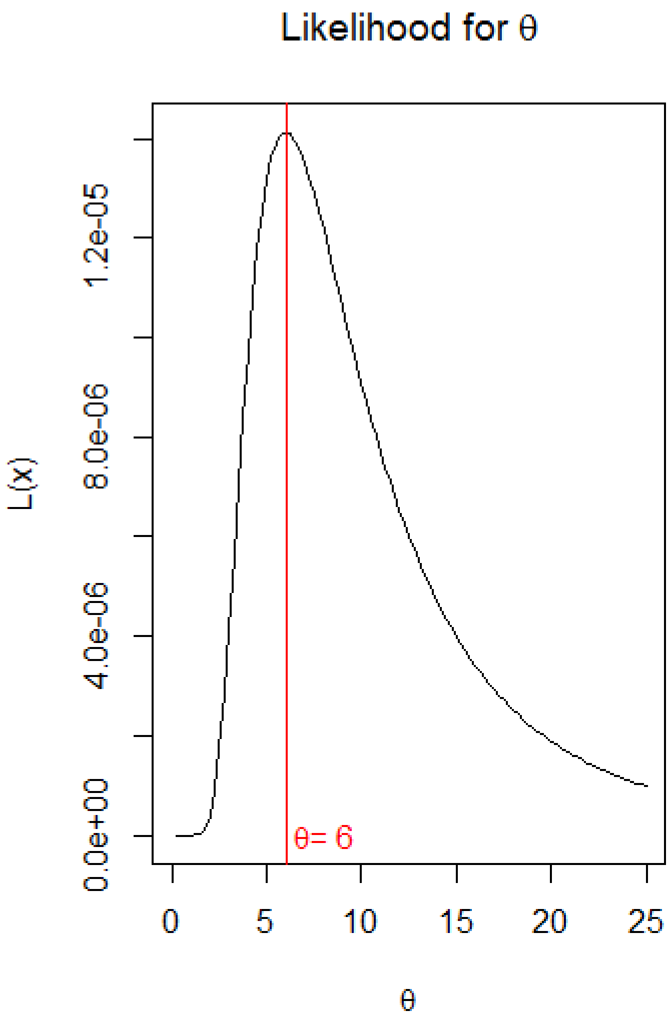
Likelihood Maximization

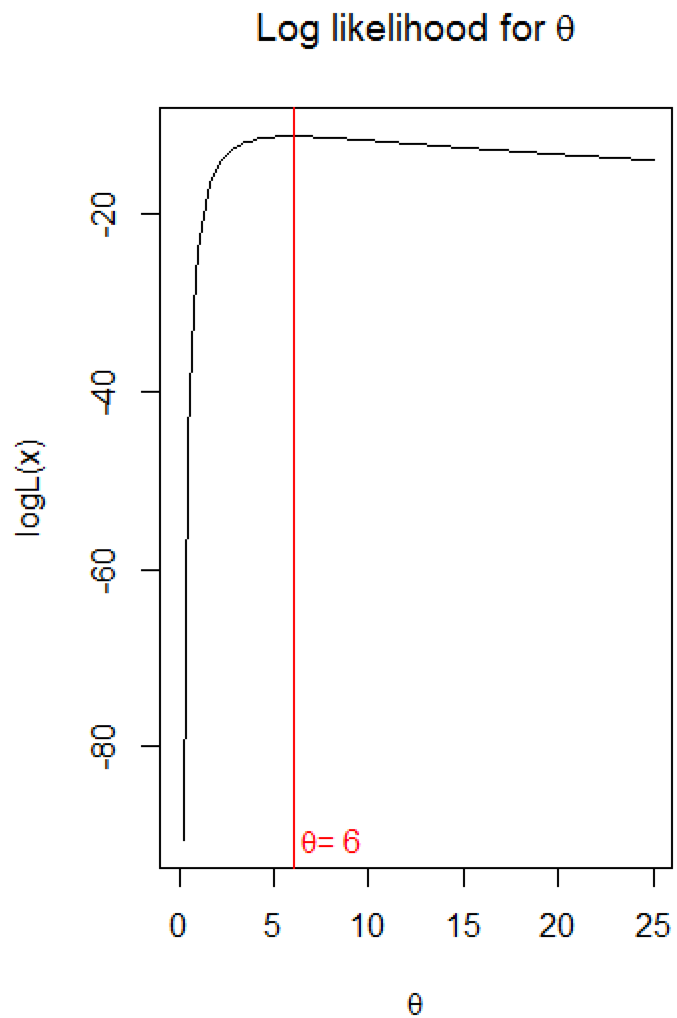
- **We can maximize the log of the likelihood**
  - $logL(\theta|\boldsymbol{x})$ or $l(\theta|\boldsymbol{x})$ is often easier to maximize
  - Since log is monotone and 1-1, maximizing $l(\theta|\boldsymbol{x})$ is equivalent to maximizing $L(\theta|\boldsymbol{x})$
  - Many common distributions belong to the *exponential family*
    - » Can be written as $f(x|\theta) = h(x)c(\theta)\exp\{w(\theta)t(x)\}$
  - Taking the log gives a much simpler expression: $\prod_{i=1}^{n} f(x|\boldsymbol{\theta})$ turns into $\sum w(\theta)t(x)$
  - Optimization via software much quicker in this form
- **Optimization using software**
  - Programs like R, JMP, and Matlab (and others) can find maxima and minima for functions that are difficult to solve in closed form
  - Plots of the likelihood give a visual representation which can "ball park" the max

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Plotting the Likelihood Function





For multiple parameters, imagine a multi-dimensional surface to be maximized

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

MLEs for Likelihoods with multiple parameters

- **The approach for likelihoods with multiple parameters is the same way.**
  - We solve

$$\frac{\partial L}{\partial \theta_1} = 0$$
$$\frac{\partial L}{\partial \theta_2} = 0$$
$$\vdots$$
$$\frac{\partial L}{\partial \theta_p} = 0$$

  simultaneously.

- **Solving these maximization problems can often be done analytically for well-understood distributions**
  - Numerical methods can also be employed

- **Homework assignment: solve for β_0 and β_1 for simple linear regression.**

$$y_i = \beta_0 + \beta_1 x_{i1} + e_i$$

  - (hint: the pdf in the likelihood is the normal distribution for the e_i's)

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Why Maximum Likelihood?

- **Intuitive**
  - Choosing parameter that is most plausible given the data is a natural approach
  - In some common problems, MLEs are often identical to familiar estimators from other methods (*e.g.*, OLS estimators from regression)

- **Broad applicability**
  - MLE Framework can be applied in a variety of scenarios
    - » Very general
  - If we can write down the likelihood, we can maximize it
  - Closed form solutions in many cases
  - Numeric solutions can be found in others

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Why Maximum Likelihood?

- **Consistency- As sample size increases, $\hat{\theta}$ converges to θ.**
- **Asymptotic Normality- As sample size increases, the distribution of $\hat{\theta}$ converges to a normal with known variance**
  – Very useful for stuff like confidence intervals
  – CAUTION: Doesn't always work well for small samples
- **Invariant to transformation- The MLE for $g(\theta)$ is $g(\hat{\theta})$.**
  – This makes estimating functions of parameters very straightforward
  – We can also find the variance (and hence confidence intervals) of functions of parameters easily using the Delta Method
    » *e.g.*, compute probability of mission success based on mean time to failure(detection)
- **Sufficiency- Contains all information in the data relevant to the parameter being estimated**

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Likelihood for Censored Data

- **We construct our Likelihood function based on the desire to use censored data:**

parameters TBD *data* *parameters TBD* *data*

$$L = \prod_{i=1}^{\#\,data\,points} [PDF(\boldsymbol{\mu}, \boldsymbol{\sigma} | t_i)]^{(1-\delta_i)} \times [1 - CDF(\boldsymbol{\mu}, \boldsymbol{\sigma} | t_i)]^{\delta_i}$$

Non-Censored data (δ_i = 0) provide information to define the shape of the PDF!

Censored data (δ_i = 1) provide information to define the shape of the CDF!

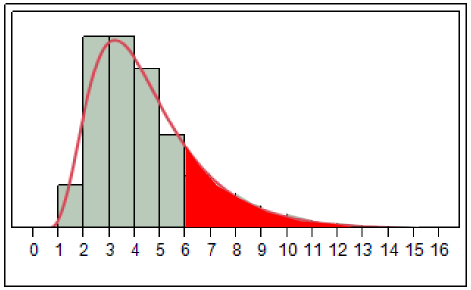
Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

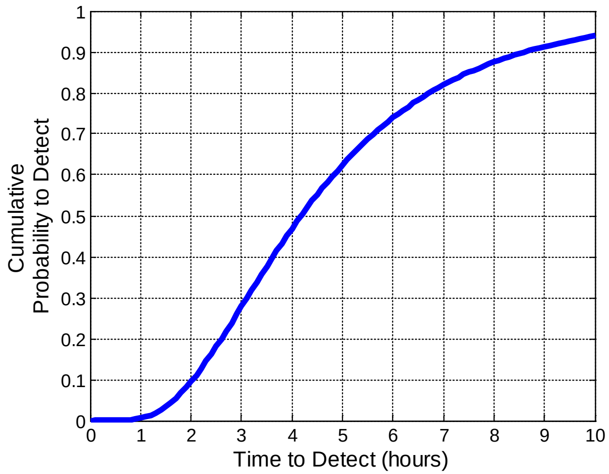
Conceptualizing the Censored-Data Fit

- **For non-censored measurements, the PDF fit is easy to conceptualize**

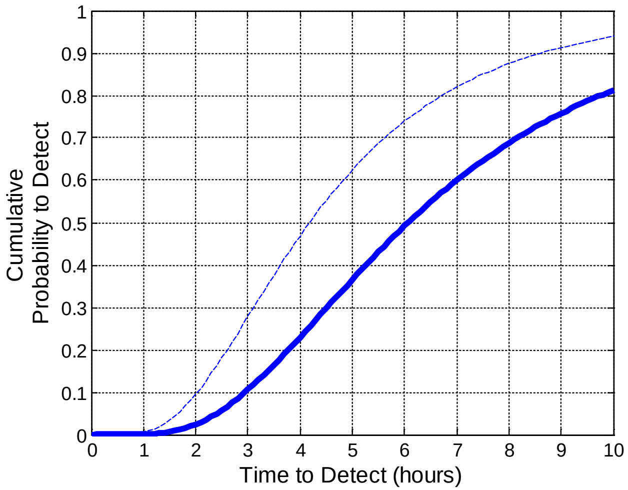
- **For censored measurements, the data can't define the PDF, but we know they contribute to the probability density beyond the censor point**



- **Example event from an OT: Time > 6 hours – that data point cannot increase the probability to the left of t=6.0 in the CDF!**
  - Detect will occur at some time in the future, so it must contribute to the probability beyond t=6.0



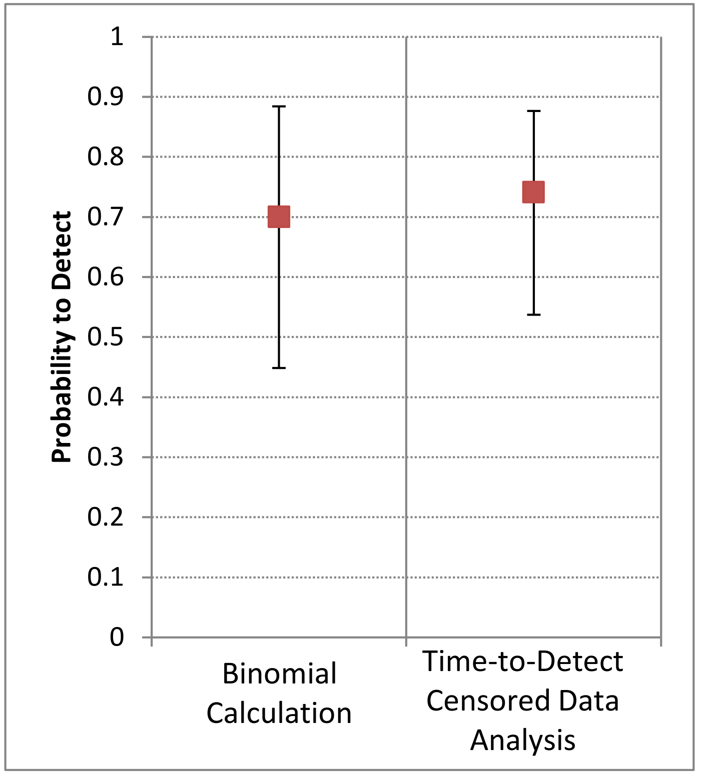
Including a bunch of censored (Time > 6 hour) events will push the CDF to the right (see how probability to detect is lower at 6 hours)

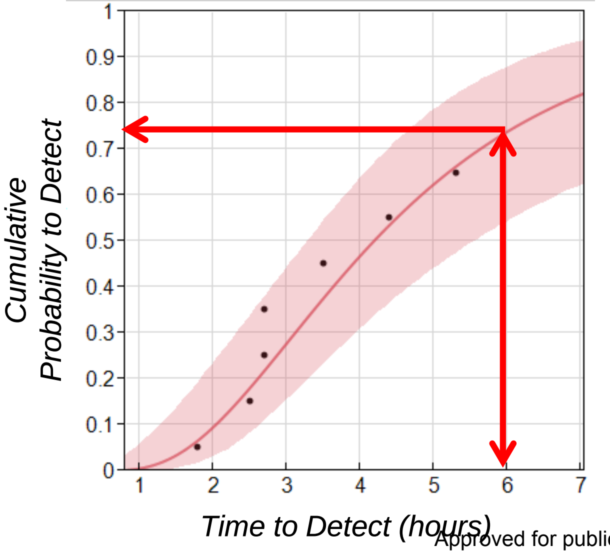


Approved for public release; distribution is unlimited.

Simplest Example

- **Consider data from slide 6…..**
- **With only 10 data points, the censored data approach provides smaller confidence intervals**
  - 16% reduction in interval size
  - Better estimate of the probability to detect
- **More confident system is meeting requirements, but with same amount of data**





	Binomial Probability Calculation	Time-to-Detect Censored Data Analysis
Confidence Threshold $P_{detect} > 0.5$ is met	82%	93%

- **Example in JMP….**
- **Data:**

Run No.	Time of Detection (hours after COMEX)
1	**4.4**
2	**2.7**
3	**>6.1**
4	**2.5**
5	**3.5**
6	**5.3**
7	**>6.2**
8	**>5.8**
9	**1.8**
10	**2.7**

Approved for public release; distribution is unlimited.

Characterizing Performance

- **Now data from a test with factors varied (DOE)…**

- **Consider a test with 16 runs**
  - **<u>Two</u>** factors examined in the test
  - Run Matrix:

	Target Fast	Target Slow	Totals
Test Location 1	4	4	8
Test Location 2	4	4	8
	8	8	**16**

  - Detection Results:

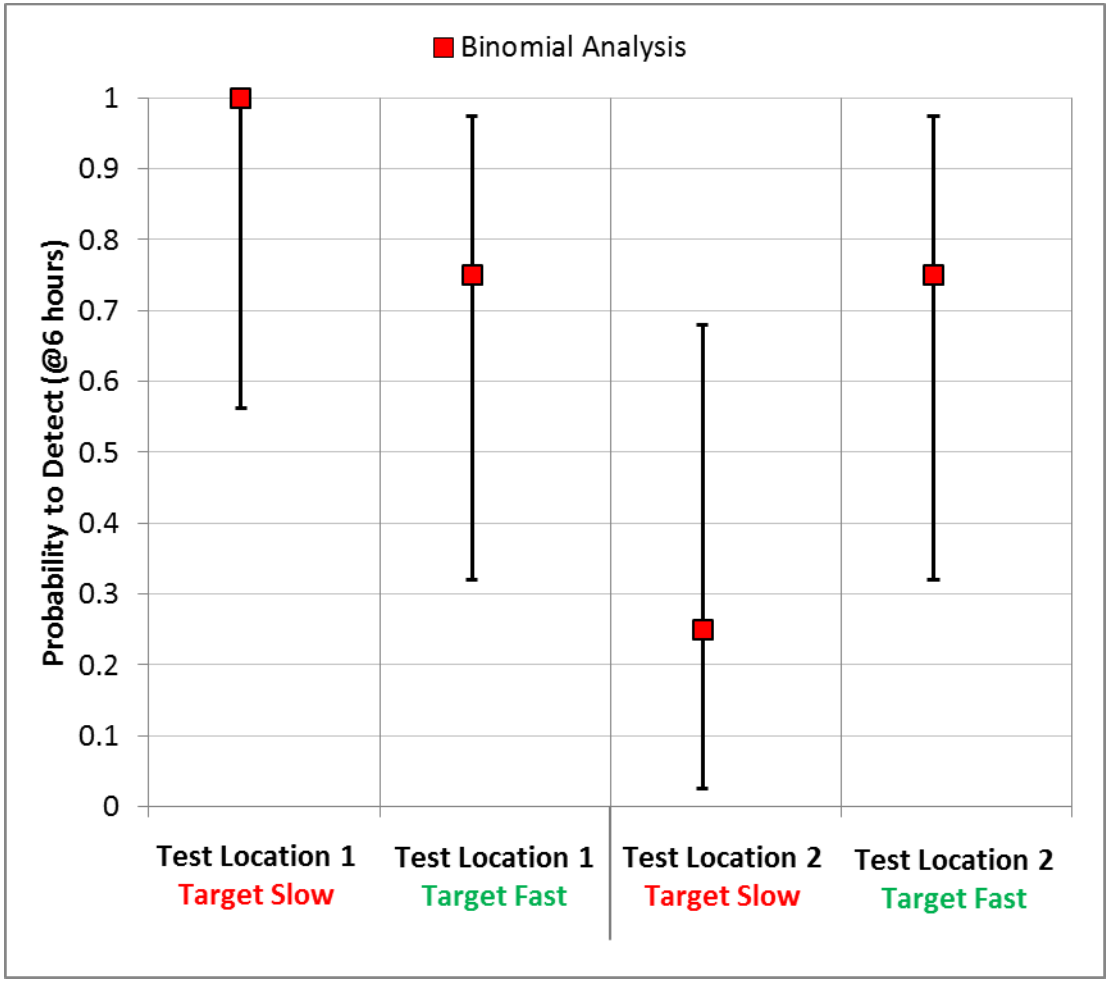
	Target Fast	Target Slow	Totals
Test Location 1	3/4	4/4	**7/8 (0.875)**
Test Location 2	3/4	1/4	**4/8 (0.5)**
	6/8 (0.75)	**5/8 (0.63)**	

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Attempt to Characterize Performance

- **As expected, 4 runs in each condition is *insufficient* to characterize performance with a binomial metric**
- **Cannot tell which factor drives performance or which conditions will cause the system to meet/fail requirements**
- **Likely will only report a 'roll-up' of 11/16**
  - 90% confidence interval: [0.45, 0.87]



Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Demo #2

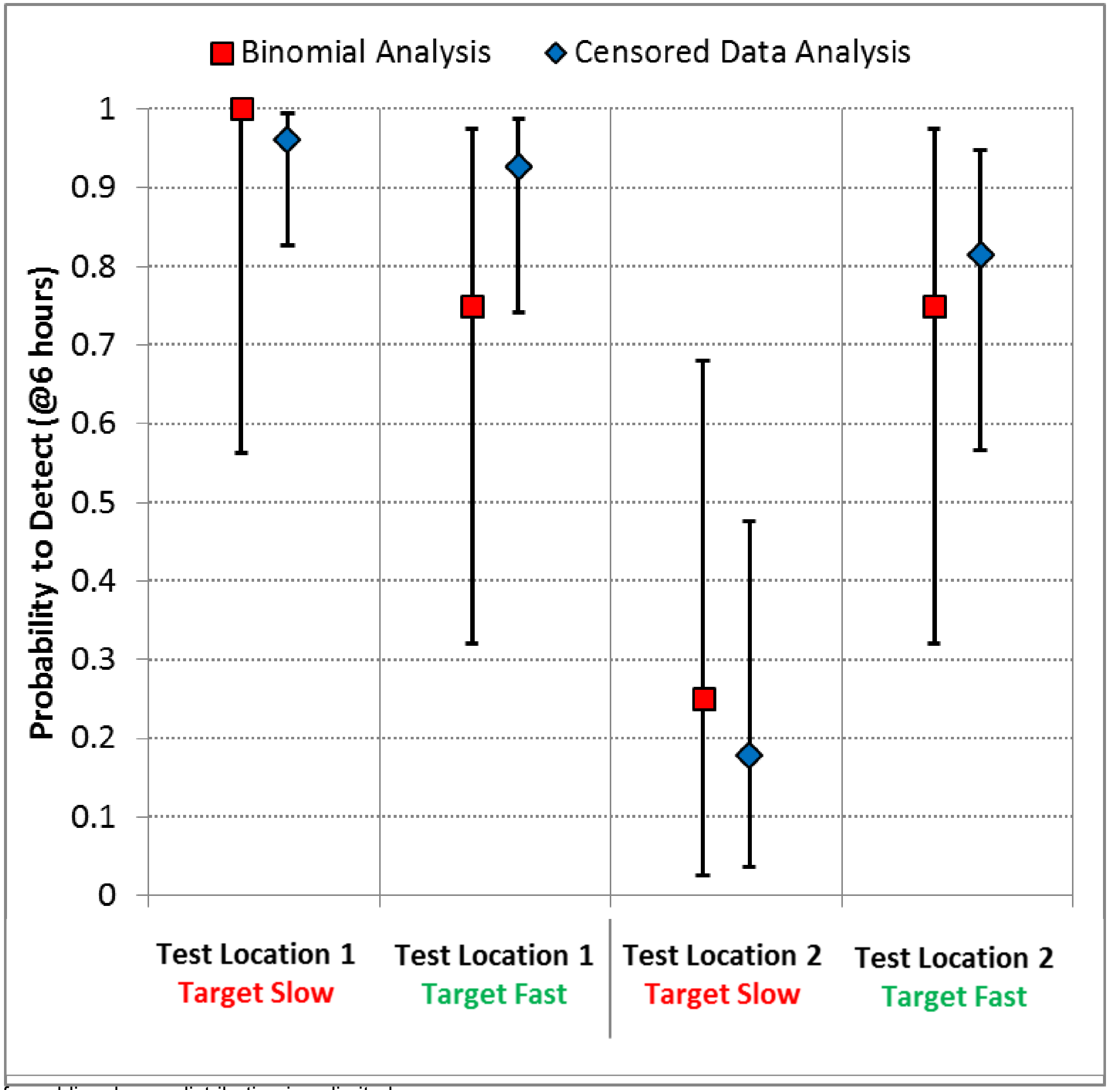
- Demo in JMP
- Data in Excel…. Analysis in JMP….. Plots in Excel…

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Characterizing Performance Better

- **Measure *time-to-detect* in lieu of binomial metric, employ censored data analysis…**
- **Significant reduction in confidence intervals!**
  - Now can tell significant differences in performance
    - » E.g., system is performing **poorly** in Location 2 against slow targets
  - We can confidently conclude performance is above threshold in three conditions
    - » Not possible with a "probability to detect" analysis!



Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

A Slight Diversion….

How did we get those confidence intervals?

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Confidence Intervals on Functions

- **Typically we are not interested in reporting confidence intervals on the parameters estimated**
- **Rather we are interested in estimating some function of those parameters**

"But the β's aren't helpful… I need to know how well the system performs against countermeasures and slow targets."
Or
"The β's don't tell me if the system passed the requirement."

	Slow Speed Target	Fast Speed Target
Location 1	a	b
Location 2	c	d

- **To construct confidence intervals on functions of parameters we need to propagate the error from the parameters to the actual response using the <u>multivariate delta method.</u>**

$$\mu = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_{12} x_{12}$$

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Delta Method: Propagation of Error

- **Earlier, we mentioned that MLEs were invariant to transformation**
  - Also true for multivariate data
  - To compute the variance, we need to use the Delta Method

- **The variance of is $\hat{\theta}$ given by the variance-covariance matrix**

$$\hat{\Sigma}_{\hat{\theta}} = \left[- \frac{\partial^2 \ell(\theta)}{\partial \theta \partial \theta^T} \right]^{-1}$$

AKA Fisher information matrix

- **Propagation of error for multiple parameters:**

$$var\left(g(\widehat{\boldsymbol{\theta}})\right) \cong g'(\widehat{\boldsymbol{\theta}})^{\mathrm{T}} \hat{\Sigma}_{\widehat{\theta}} \, \mathrm{g}'(\widehat{\boldsymbol{\theta}})$$, where $$g'(\widehat{\boldsymbol{\theta}}) = \begin{pmatrix} \frac{\partial g}{\partial \theta_1} \\ \vdots \\ \frac{\partial g}{\partial \theta_p} \end{pmatrix}$$

Derivative first, then evaluate at MLE estimates for parameters

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Delta Method: Propagation of Error

- **Let's look at the case of two parameters: g(θ)=g(μ,σ)**
- **A (1-α)100% confidence interval on g(μ,β) is:**

$$[g_L, g_u] = \hat{g}\,(\hat{\mu}, \hat{\sigma}) \pm z_{(1-\alpha/2)}\sqrt{var\big(g(\hat{\mu}, \hat{\sigma})\big)}$$

Note this assumes normality in the parameters

- **Where**

$$var(\hat{g}(\hat{\mu},\hat{\sigma})) = \left[\left(\frac{\partial g}{\partial \mu}\right)^2 V\hat{a}r(\hat{\mu}) + \left(\frac{\partial g}{\partial \sigma}\right)^2 V\hat{a}r(\hat{\sigma}) + 2\left(\frac{\partial g}{\partial \mu}\right)\left(\frac{\partial g}{\partial \sigma}\right) C\hat{o}v(\hat{\mu},\hat{\sigma})\right]$$

Take derivative first, then plug in MLE values for parameters

Most software will output the covariance matrix from the MLE fit

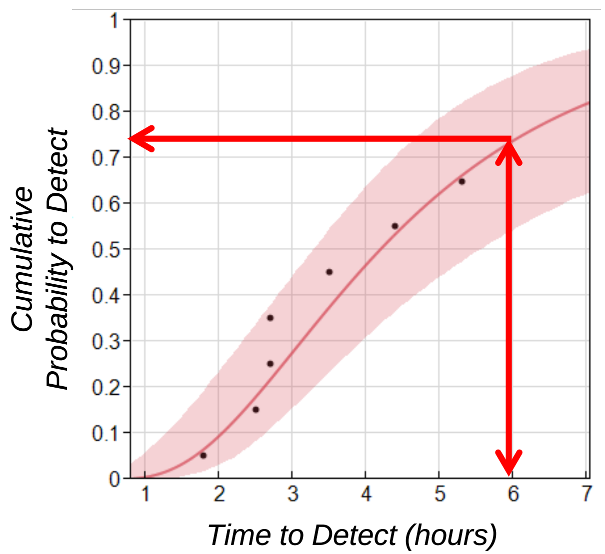
Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Example Applications of Delta Method

At the censor point, or some other point of interest

- **In censored data analyses we are often interested in:**
  - $\hat{g}\,(\hat{\mu},\hat{\sigma}) = \mathrm{CDF}(t = t_0, \hat{\mu}, \hat{\sigma})$
  - We can now estimate the upper and lower confidence intervals on $P_{detect}(t = 6 \text{ hours})$, where for the lognormal fit we use:
  - $\hat{g}\,(\hat{\mu},\hat{\sigma}) = \Phi\left(\frac{\ln(6)-\hat{\mu}}{\hat{\sigma}}\right) = 0.5 \cdot \mathrm{erfc}\left(-\frac{\ln(6)-\hat{\mu}}{\hat{\sigma}\sqrt{2}}\right)$



- **Another common quantity of interest is the quantile function:**
  - $\hat{g}\,(\hat{\mu},\hat{\sigma}) = t_p = \exp(\hat{\mu} + \Phi^{-1}(p) * \hat{\sigma})$
  - We can now estimate upper and lower confidence bounds for the 50th percentile of the time distribution (median time to detect)

Log-normal quantile function

Approved for public release; distribution is unlimited.

Delta Method: Data with Factors

- **If we are determining the effect of factors and want to estimate Probability of [Detect/Hit/etc.] in Condition 1, 2, 3:**

$$\hat{g}\,(\hat{\mu},\hat{\sigma}) = \Phi\left(\frac{\ln(6)-\hat{\mu}}{\hat{\sigma}}\right) = 0.5\cdot\operatorname{erfc}\left(-\frac{\ln(6)-\hat{\mu}}{\hat{\sigma}\sqrt{2}}\right)$$

Where $\hat{\mu} = \widehat{\beta_0} + \widehat{\beta_1}x_1 + \widehat{\beta_2}x_2 + \widehat{\beta_{12}}x_{12} + \ldots$ or $\hat{\mu} = \mathbf{X}\cdot\boldsymbol{\beta}$

- **Still use same approach for estimating confidence intervals on $\hat{g}\,(\hat{\mu},\hat{\sigma})$ -- now just include full model, and plug in x_i's for point of interest in factor space (after the derivatives)**

$$var\left(g(\widehat{\boldsymbol{\theta}})\right) \cong g'(\widehat{\boldsymbol{\theta}})^{\mathrm{T}}\,\hat{\Sigma}_{\hat{\theta}}\,g'(\widehat{\boldsymbol{\theta}}) \quad \text{, where} \quad g'(\widehat{\boldsymbol{\theta}}) = \begin{pmatrix} \frac{\partial g}{\partial\theta_1} \\ \vdots \\ \frac{\partial g}{\partial\theta_p} \end{pmatrix}$$

	Slow Speed Target	Fast Speed Target
Location 1	(+1, +1)	(-1,+1)
Location 2	(+1,-1)	(-1,-1)

Approved for public release; distribution is unlimited.

Full Example: Submarine Detection Time

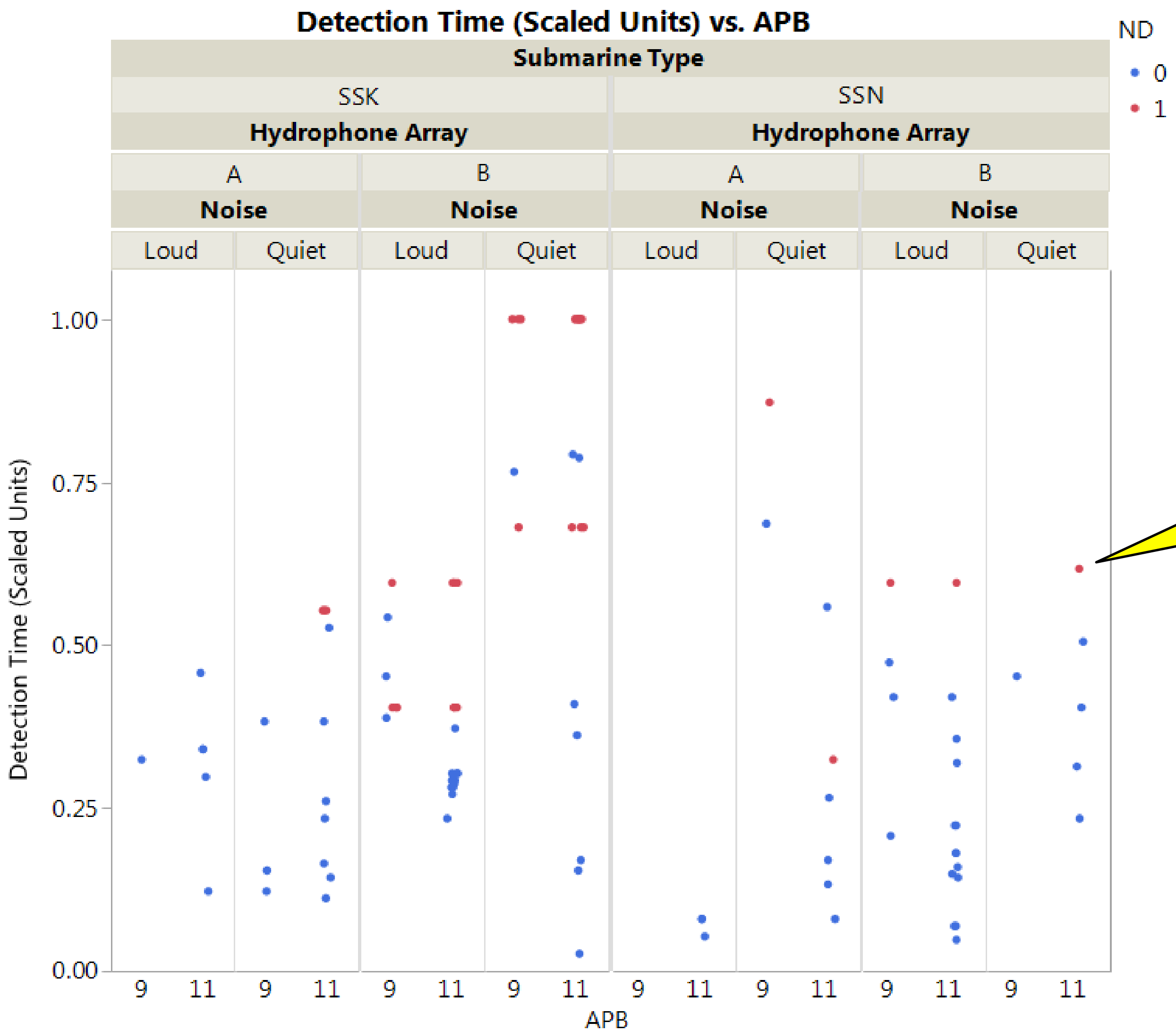
- **System Description**
  - Sonar system replica in a laboratory
  - Data recorded during real-world interactions can be played back in real-time.
  - System can process the raw hydrophone-level data with any desired version of the sonar software.
  - Upgrade every two years; test to determine if new version is better
  - Advanced Processor Build (APB) 2011 contains a potential advancement over APB 2009 (new detection method capability)

- **Response Variable:** Detection Time
  - Time from first appearance in recordings until operator detection
    - » Failed operator detections resulted in *right censored data*
- **Factors:**
  - Operator proficiency (quantified score based on experience, time since last deployment, etc.)
  - Submarine Type (SSN, SSK)
  - System Software Version (APB 2009, APB 2011)
  - Array Type (A, B)
  - Target Loudness (Quiet, Loud)

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Look at the Data First



"Non-detects" or censored data are indicated as "1" for ease in coding in software

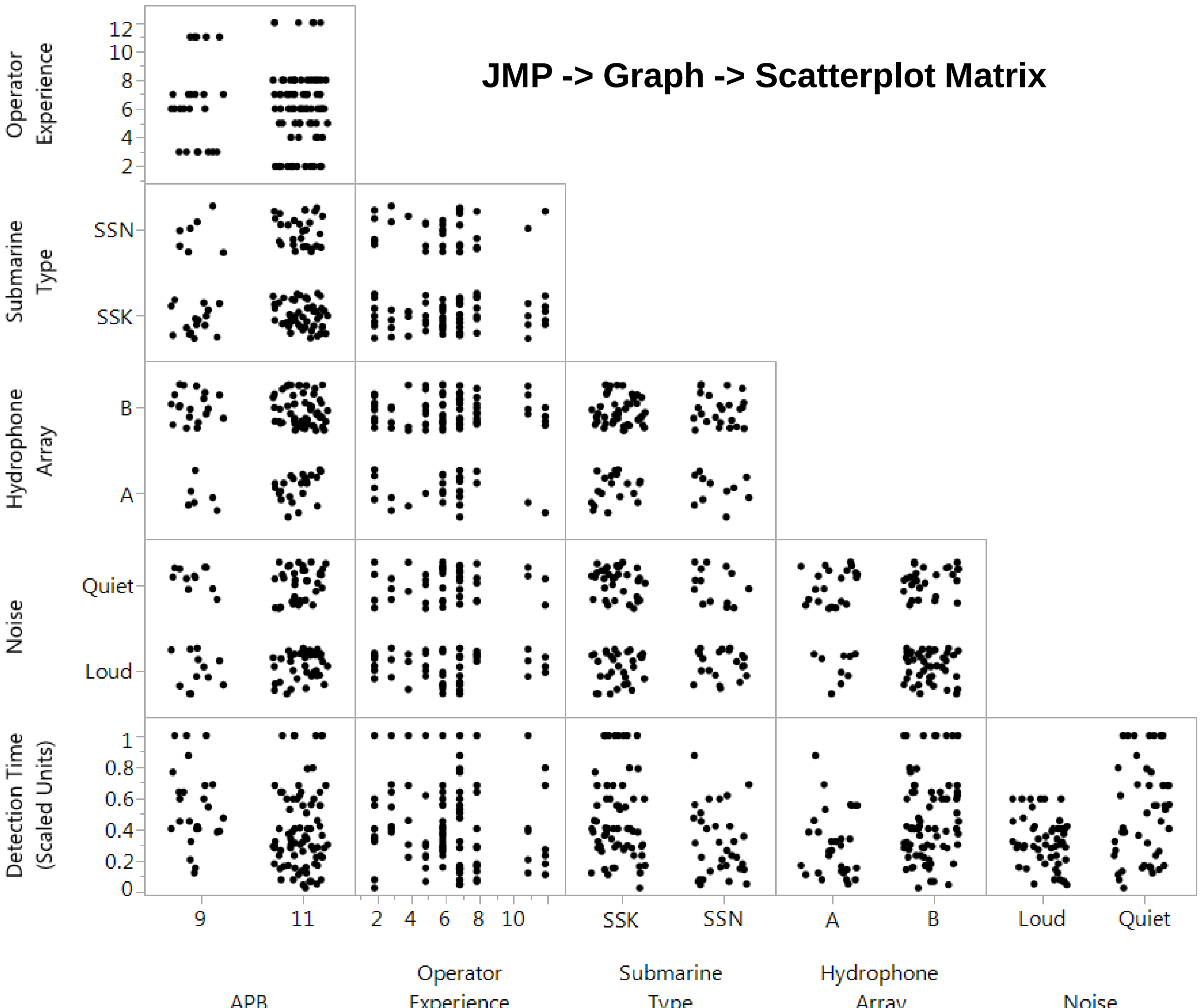
Note some events ended without a successful detection

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Data Examined

JMP -> Graph -> Scatterplot Matrix



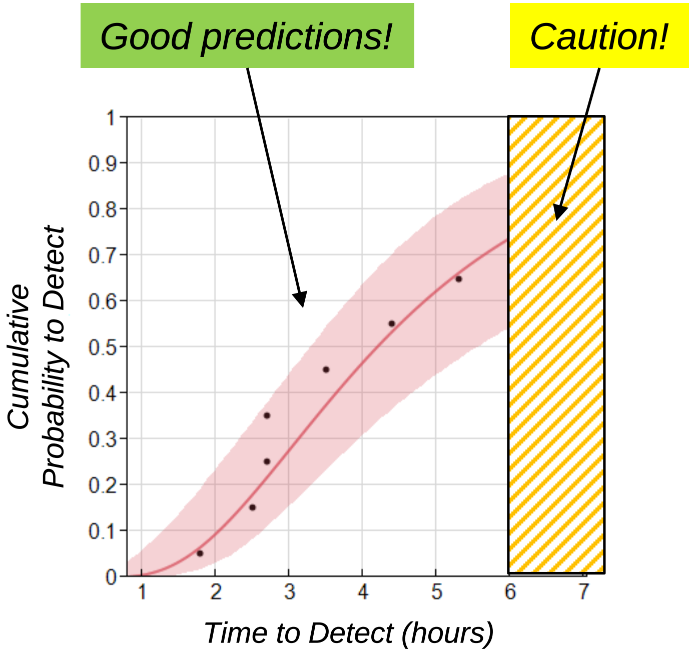
Apply standard model selection techniques:

- Ensure factors are not strongly correlated or significant missing data that would preclude fitting terms
- Holes in the data mean some model terms will be inestimable!

Approved for public release; distribution is unlimited.

Key Assumption

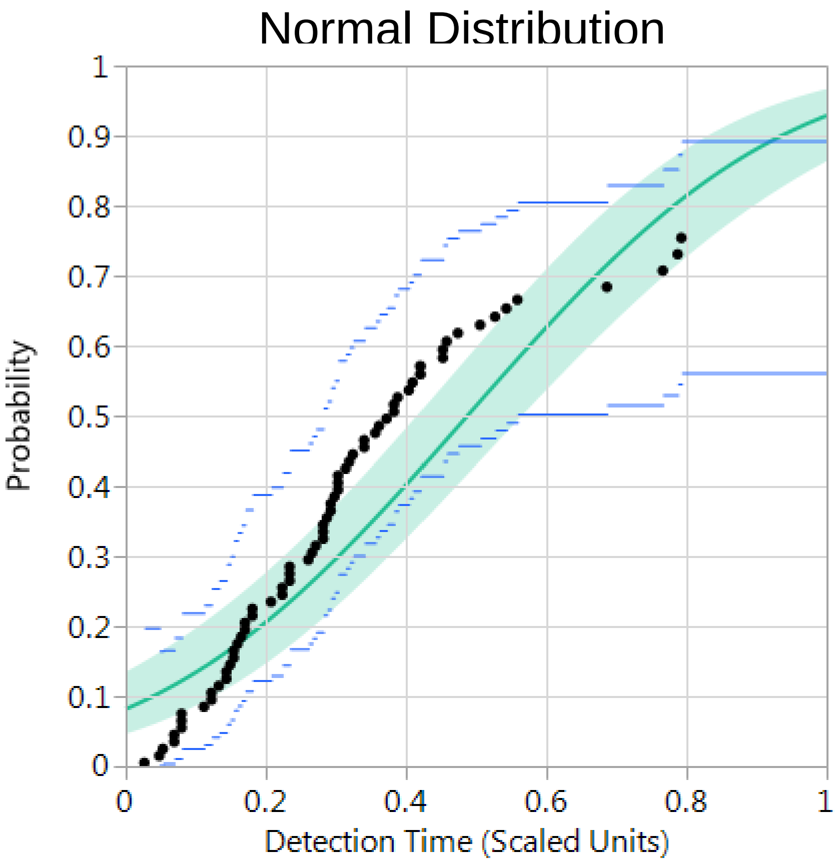
- **Right-censored data assume that detection will occur eventually, if time continues**
- **Put another way: this approach assumes the cumulative probability to succeed (detect, hit, etc.) is monotonically increasing**
- **Often a stumbling block to employing the method, esp. when the chance to detect/hit/succeed after a certain point is physically zero**
  - Example: target leaves the area providing no further detection opportunities
- **Can still use the method!**
  - Should not predict beyond the censor point, and generally should stay well-inside
  - If possible, structure the test so that censor point is well-outside where you need to predict/estimate performance

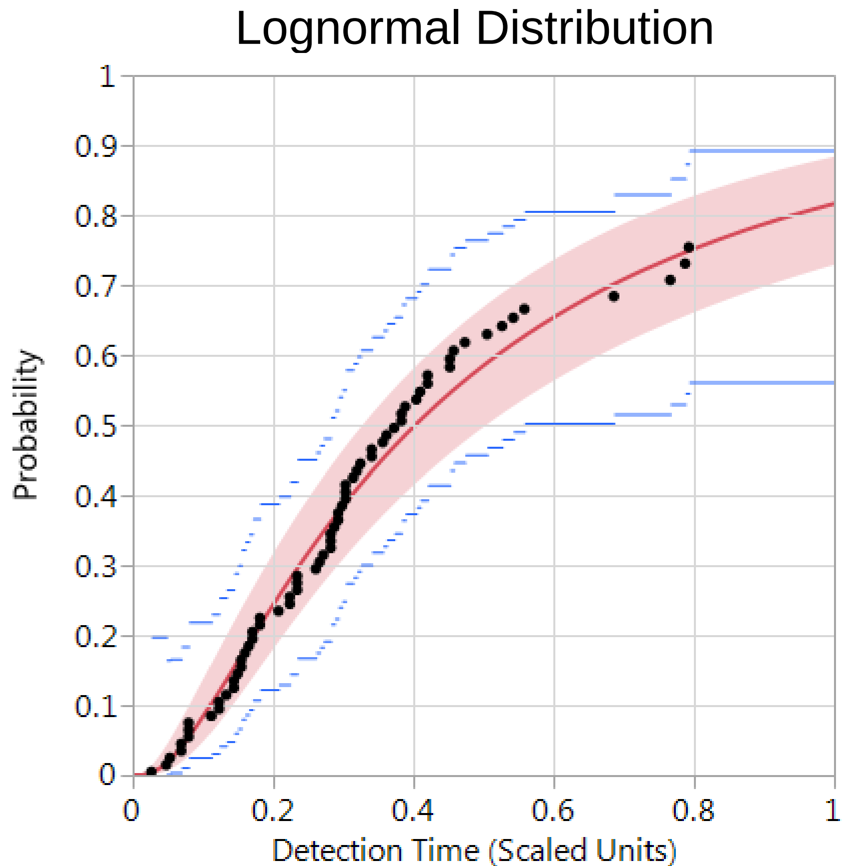


Approved for public release; distribution is unlimited.

Detection Time Distribution

- **Recall: goal is to determine PDF/CDF that accurately reflects the data**
  - Detection time does not follow a normal distribution

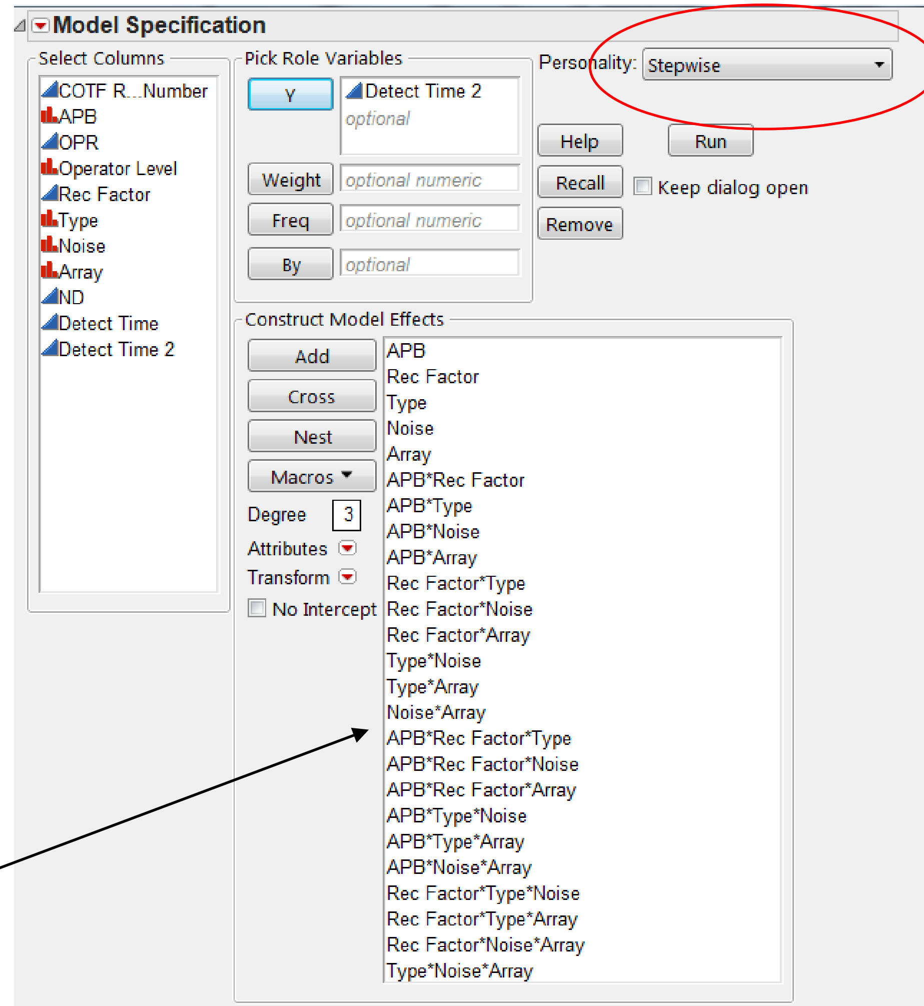




Approved for public release; distribution is unlimited.

Model Selection

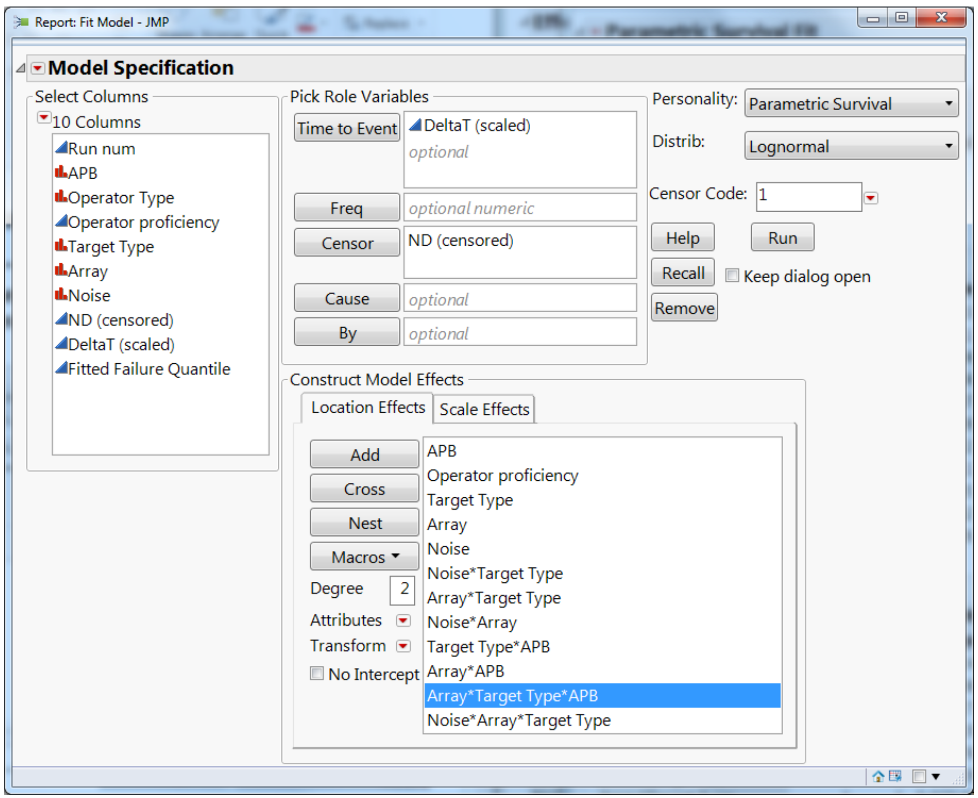
- **Start with automated model selection**
  - Assumes normality, and no censoring, but factor significance is fairly robust to this assumption
  - Will narrow down set of factors to a manageable number



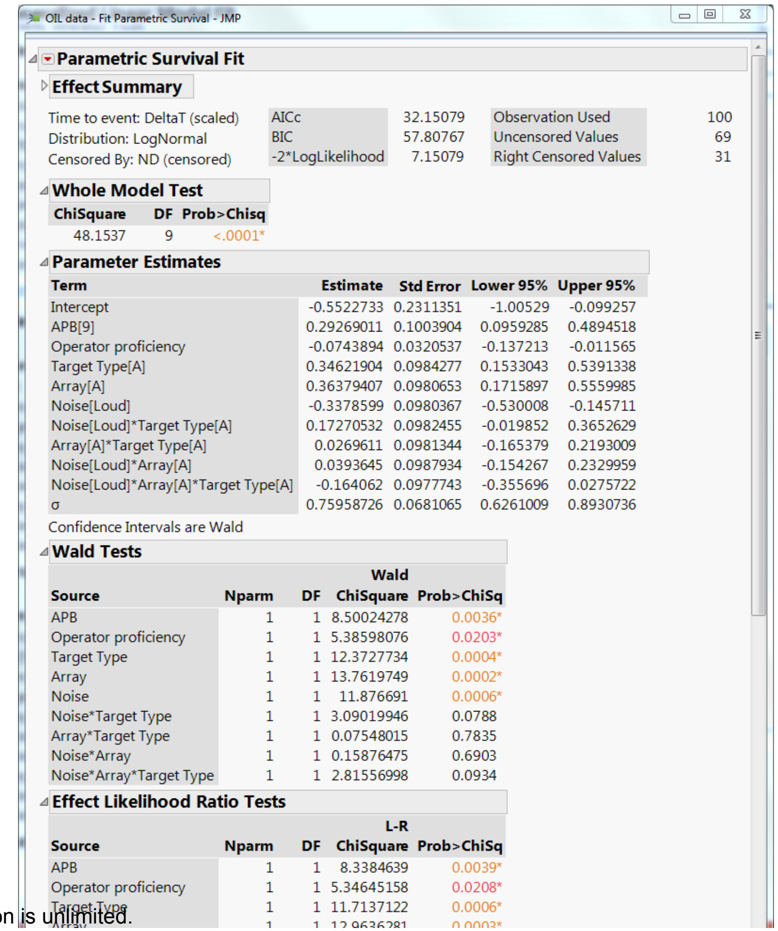
For this case, we choose to start with all possible terms up to 3-way interactions

Sub Detection Example: Model Selection

- **Fit lognormal model using down-selected factors from automated results**



- **Further reduce model by hand**
  - Remove one term at a time based on p-value



Approved for public release; distribution is unlimited.

ΔT Final Model

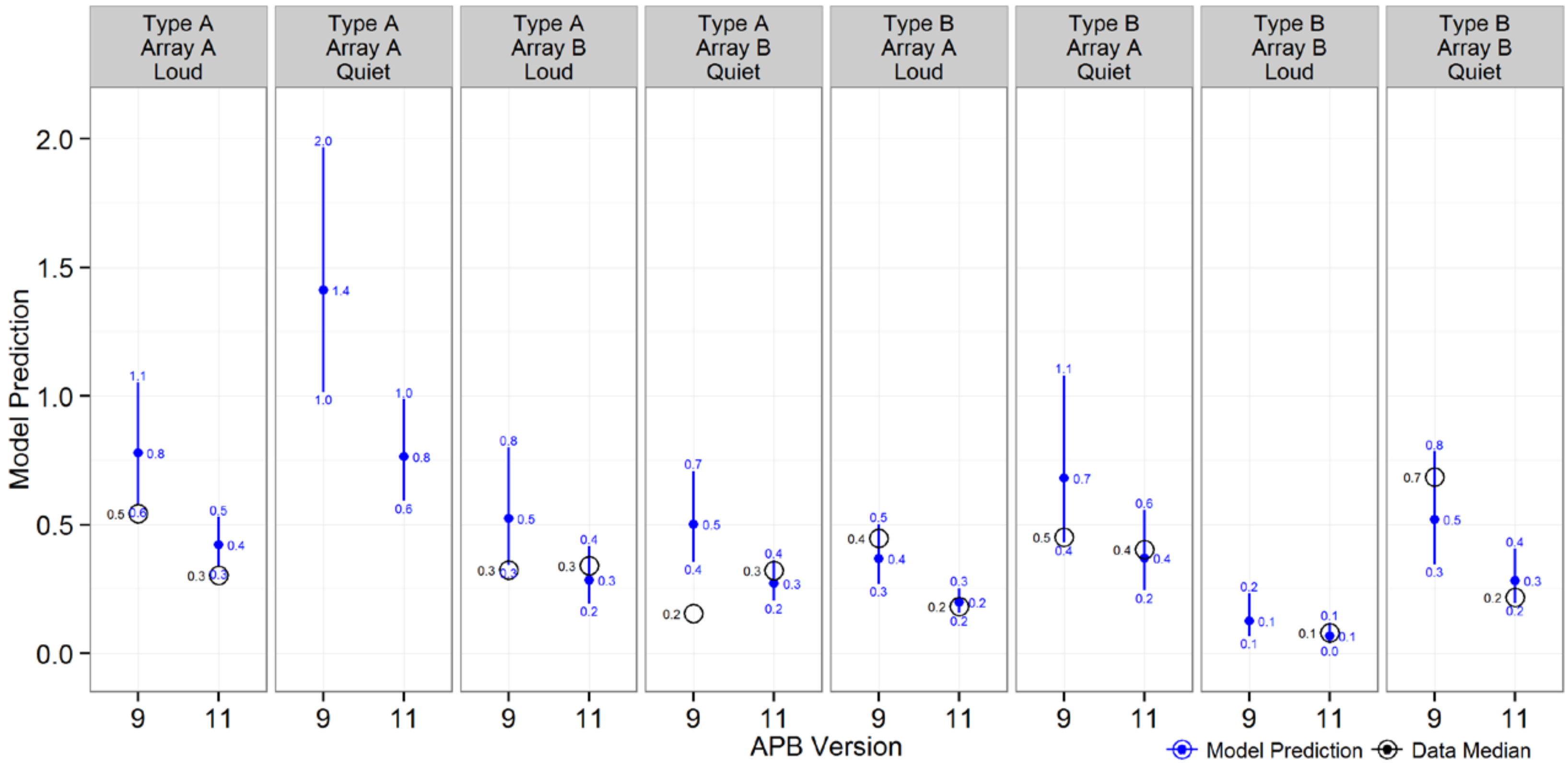
$$\mu = \beta_0 + \beta_1\, RF + \beta_2\, APB + \beta_3\, Target + \beta_4\, Noise + \beta_5\, Array + \beta_6\, Target * Noise + \beta_7\, Target * Array + \beta_8\, Noise * Array + \beta_9\, Target * Noise * Array$$

Term	Description of the Effect	p-Value
β_1 (RF)	Increased recognition factors resulted in shorter detection times	0.0227
β_2 (APB)	Detection time is shorter for APB-11	0.0025
β_3 (Target)	Detection time is shorter for Type B targets	0.0004
β_4 (Noise)	Detection time is shorter for loud targets	0.0012
β_5 (Array)	Detection time is shorter for the Type B array	0.0006
β_6 (Target*Noise)	Additional model terms added to improve predictions. Third order interaction is marginally significant. Therefore, all second order interactions nested within the third order interaction were retained to preserve model hierarchy.	0.0628
β_7 (Target*Array)		0.9091
β_8 (Noise*Array)		0.8292
β_9 (Target*Noise*Array)		0.0675

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Results and Model Validation



Median detection times show a clear advantage of APB-11 over the legacy APB

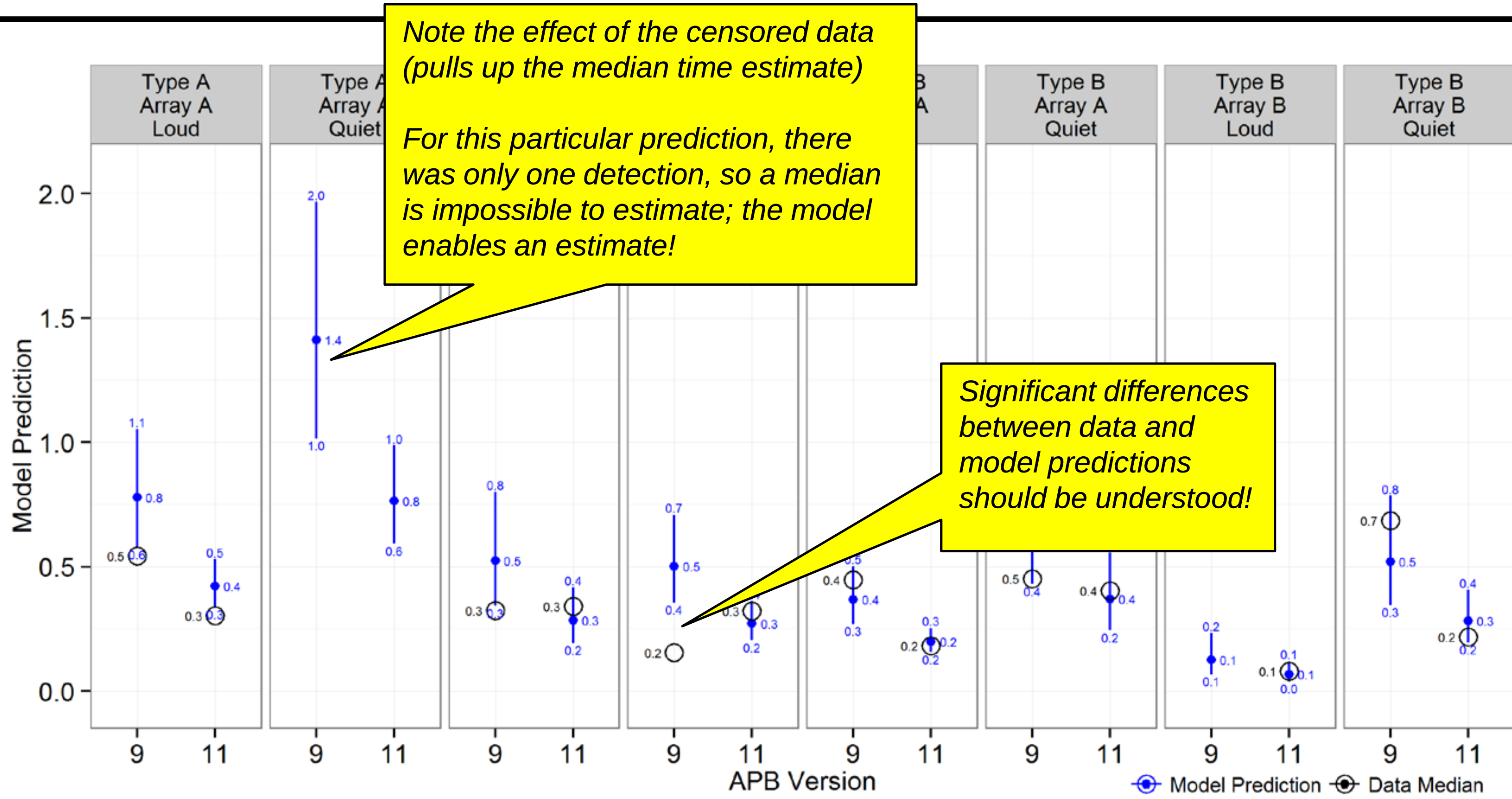
- **Check: Model prediction results are consistent with the non-parametric median estimates**

Note: Non-parametric medians calculated from the Kaplain-Meier estimator per bin and model estimates are the median with an 80% confidence interval

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Results and Benefits of Approach



- **Confidence interval widths reflect weighting of data towards APB-11**
- **Statistical model provides insights in areas with limited data**
  - Note median detection time in cases with heavy censoring is shifted higher

Note: Non-parametric medians calculated from the Kaplain-Meier estimator per bin and model estimates are the median with an 80% confidence interval

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

What if we had just done the Binomial Analysis?

Censored analysis

Effect Likelihood Ratio Tests

Source	Nparm	DF	L-R ChiSquare	Prob>ChiSq
APB	1	1	8.3384639	0.0039*
Operator proficiency	1	1	5.34645158	0.0208*
Target Type	1	1	11.7137122	0.0006*
Array	1	1	12.9636281	0.0003*
Noise	1	1	11.244081	0.0008*
Noise*Target Type	1	1	2.9921104	0.0837
Array*Target Type	1	1	0.07572392	0.7832
Noise*Array	1	1	0.15867745	0.6904
Noise*Array*Target Type	1	1	2.76745543	0.0962

Parameter Estimates

Term	Estimate	Std Error
Intercept	-0.5522733	0.2311351
APB[9]	0.29269011	0.1003904
Operator proficiency	-0.0743894	0.0320537
Target Type[A]	0.34621904	0.0984277
Array[A]	0.36379407	0.0980653
Noise[Loud]	-0.3378599	0.0980367
Noise[Loud]*Target Type[A]	0.17270532	0.0982455
Array[A]*Target Type[A]	0.0269611	0.0981344
Noise[Loud]*Array[A]	0.0393645	0.0987934
Noise[Loud]*Array[A]*Target Type[A]	-0.164062	0.0977743
σ	0.75958726	0.0681065

Logistic regression (binomial analysis)

Effect Tests

Source	DF	L-R ChiSquare	Prob>ChiSq
APB	1	1.8472374	0.1741
Operator proficiency	1	4.1665227	0.0412*
Target Type	1	0	1.0000
Array	1	2.6641235	0.1026
Noise	1	3.0706052	0.0797
Noise*Target Type	1	0	1.0000
Array*Target Type	1	0	1.0000
Noise*Array	1	0.3772702	0.5391
Noise*Array*Target Type	1	0	1.0000

Parameter Estimates

Term	Estimate	Std Error
Intercept	0.0106256	0.664936
APB[9]	-0.357354	0.2737247
Operator proficiency	0.1714661	0.0929977
Target Type[A]	-0.352271	0.3699359
Array[A]	-0.442248	0.3686938
Noise[Loud]	0.4906555	0.3686719
Noise[Loud]*Target Type[A]	0.1493408	0.3687294
Array[A]*Target Type[A]	-0.523949	0.368399
Noise[Loud]*Array[A]	-0.135902	0.3702743
Noise[Loud]*Array[A]*Target Type[A]	-0.037941	0.3682997

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

What if we had just done the Binomial Analysis?

Censored analysis

Effect Likelihood Ratio Tests

Source	Nparm	DF	L-R ChiSquare	Prob>ChiSq
APB	1	1	8.3384639	0.0039*
Operator proficiency	1	1	5.34645158	0.0208*
Target Type	1	1	11.7137122	0.0006*
Array	1	1	12.9636281	0.0003*
Noise	1	1	11.244081	0.0008*
Noise*Target Type	1	1	2.9921104	0.0837
Array*Target Type	1	1	0.07572392	0.7832
Noise*Array	1	1	0.15867745	0.6904
Noise*Array*Target Type	1	1	2.76745543	0.0962

Parameter Estimates

Term	Estimate	Std Error
Intercept	-0.5522733	0.2311[illegible]
APB[9]	0.29269011	0.1003[illegible]
Operator proficiency	-0.0743894	0.0320[illegible]
Target Type[A]	0.34621904	0.0984277
Array[A]	0.36379407	0.0980653
Noise[Loud]	-0.3378599	0.0980367
Noise[Loud]*Target Type[A]	0.17270532	0.0982455
Array[A]*Target Type[A]	0.0269611	0.0981344
Noise[Loud]*Array[A]	0.0393645	0.0987934
Noise[Loud]*Array[A]*Target Type[A]	-0.164062	0.0977743
σ	0.75958726	0.0681065

Logistic regression (binomial analysis)

Most effects no longer significant!

[illegible]ests

Source	DF	L-R ChiSquare	Prob>ChiSq
APB	1	[illegible]	0.1741
Operator proficiency	1	4.1665227	0.0412*
Target Type	1	0	1.0000
Array	1	2.6641235	0.1026
Noise	1	3.0706052	0.0797
Noise*Target Type	1	0	1.0000
Array*Target Type	1	0	1.0000
Noise*Array	1	0.3772702	0.5391
Noise*Array*Target Type	1	0	1.0000

Large errors/conf. intervals – low precision

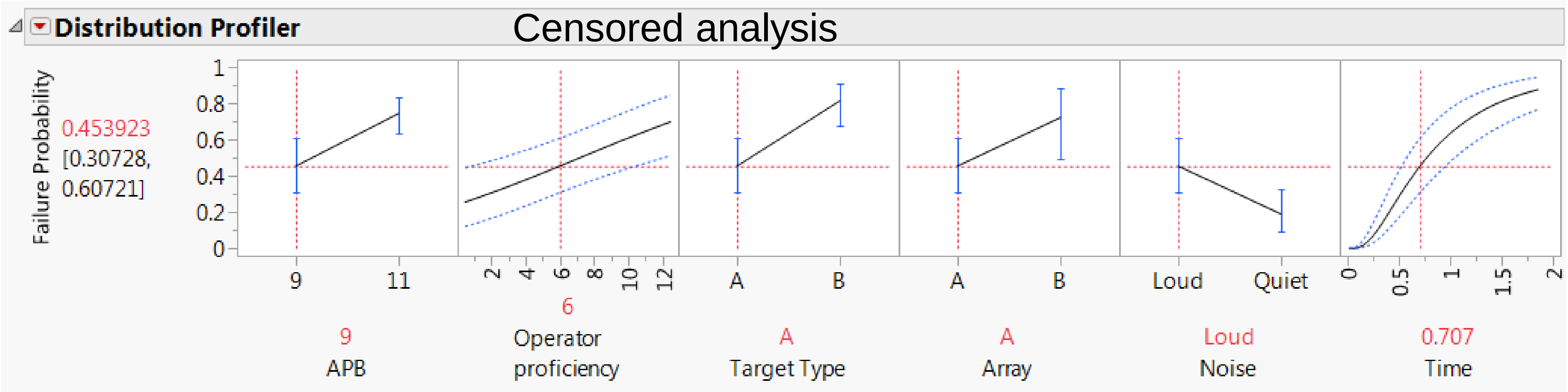
Parameter Estimates

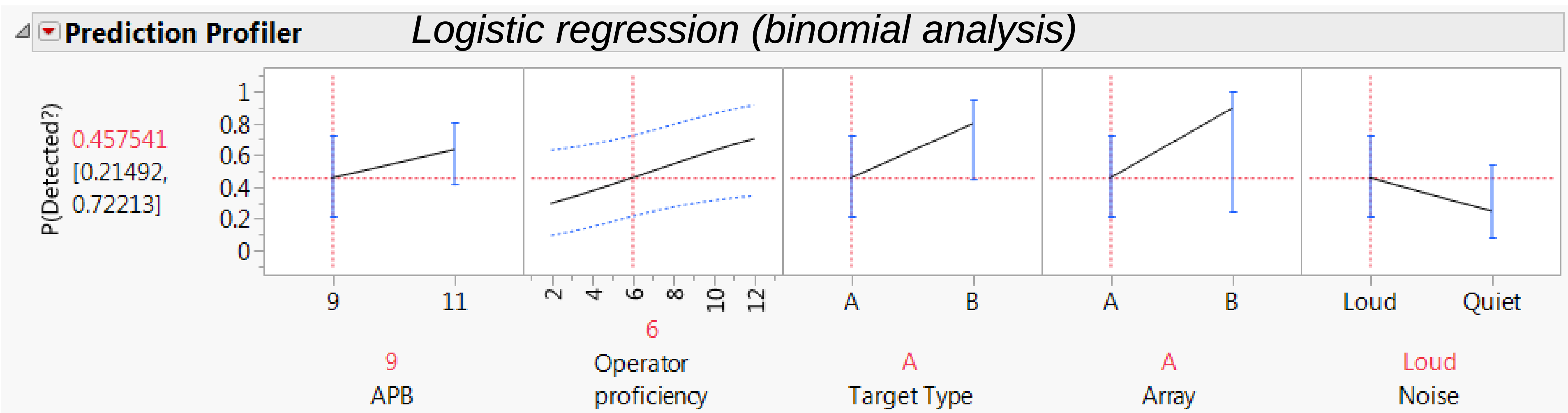
Term	Estimate	Std Error
[illegible]	0.0106256	0.664936
[illegible]	-0.357354	0.2737247
Operator proficiency	0.1714661	0.0929977
Target Type[A]	[illegible]	0.3699359
Array[A]	-0.442248	0.3686938
Noise[Loud]	0.4906555	0.3686719
Noise[Loud]*Target Type[A]	0.1493408	0.3687294
Array[A]*Target Type[A]	-0.523949	0.368399
Noise[Loud]*Array[A]	-0.135902	0.3702743
Noise[Loud]*Array[A]*Target Type[A]	-0.037941	0.3682997

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Comparing Precision of Results

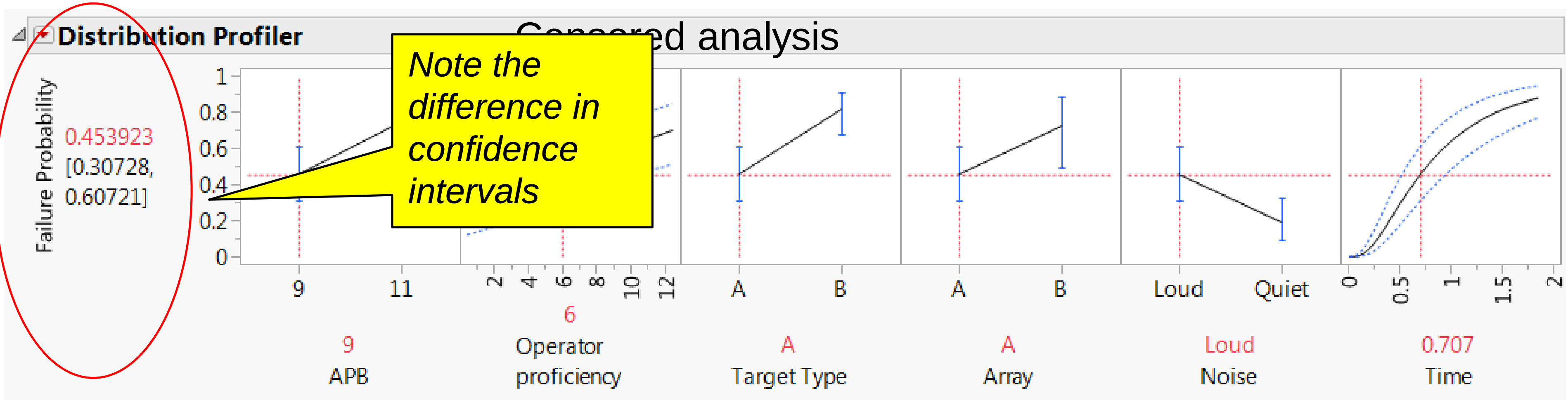




Approved for public release; distribution is unlimited.

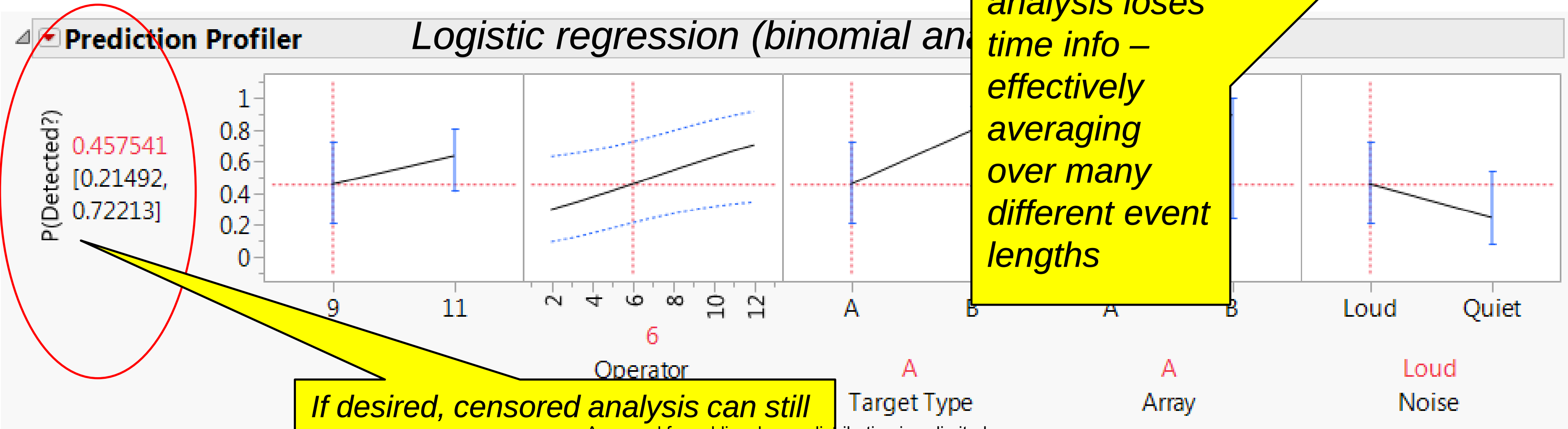
Approved for public release; distribution is unlimited.

Comparing Precision of Results



Note the difference in confidence intervals

Binomial analysis loses time info – effectively averaging over many different event lengths



If desired, censored analysis can still provide the "probability to detect"

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Power and Confidence

- **Power and confidence are only meaningful in the context of a hypothesis test**

- **Statistical hypotheses:**

H_0: *Mean Time to detect is the same in all environments*

H_1: *Mean Time to detect differs between Environment 1 and Environment 2*

$$H_0: \mu_{Env1} = \mu_{Env2}$$
$$H_1: \mu_{Env1} \neq \mu_{Env2}$$

- **Power is the probability that we conclude that the Environment (Test Location) makes a difference when it truly does have an effect.**

- **Similarly, power can be calculated for any other factor or model term**

Test Decision	*Difference*	*No Difference*
Accept H_O	*False Negative (β Risk)*	*Confidence (1-α)*
Reject H_O	*Power (1-β)*	*False Positive (α Risk)*

Real World

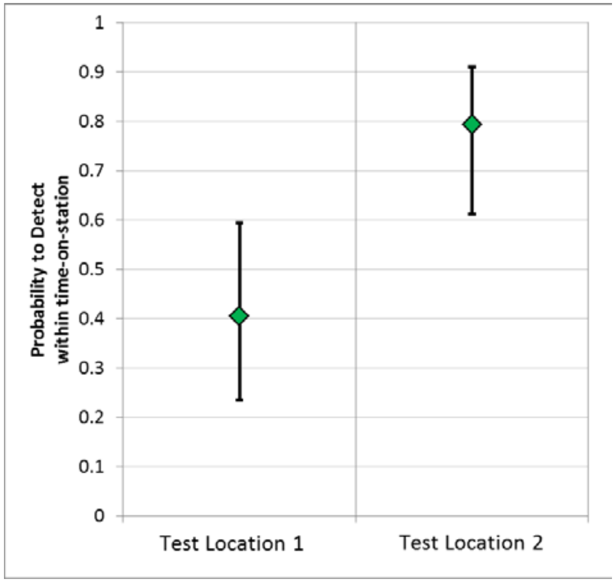
We need to understand risk!

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Sizing Tests

- **Why size a test based on ability to detect differences in P_{detect}?**
  - This is standard way to employ power calculations to detect factor effects in DOE methodology
  - We <u>are</u> interested in performance differences – this is how we *characterize performance* across the operational envelope
  - This is also how we ensure a level of precision occurs in our measurement of P_{detect} (size of the "error bars" will be determined)



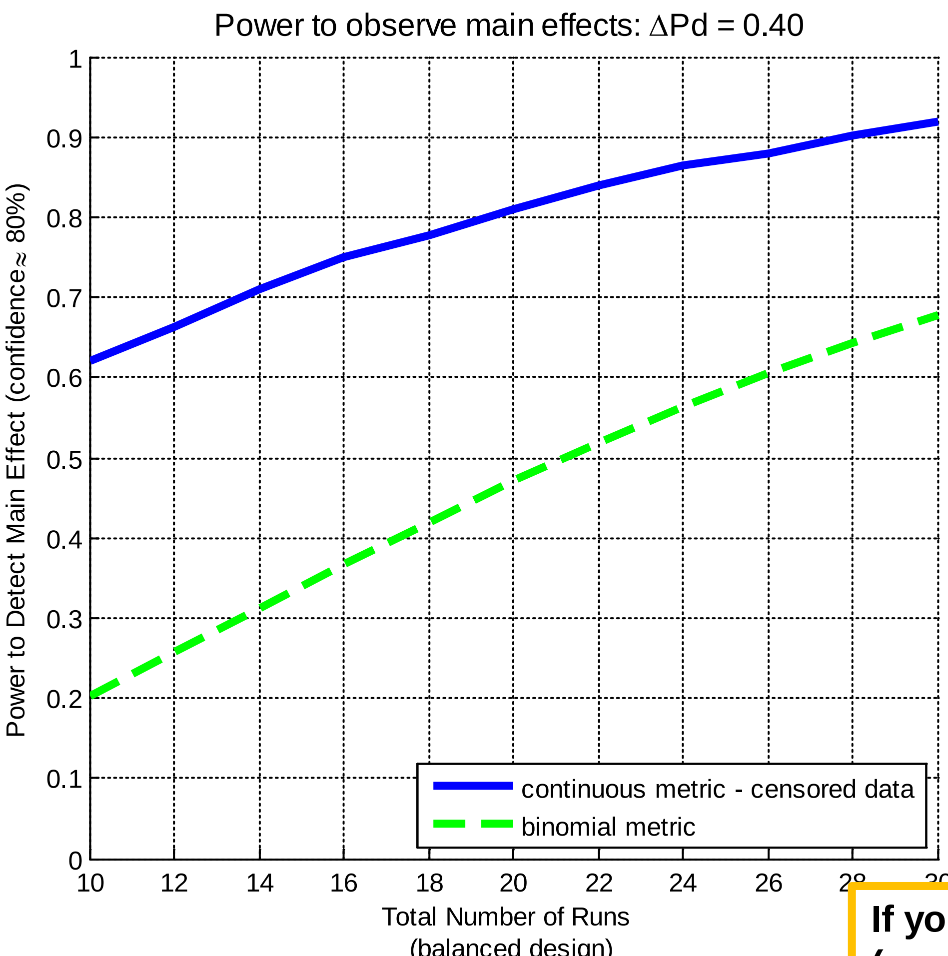
If we size the test to detect this difference, then the confidence intervals on the results will be approx. this big

If the measured delta is different than assumed, still ensure a level of accuracy in the measurement

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Sizing Tests



Total Sample Size required to detect Factor Effects with 90% confidence, 80% power

ΔP detectable	Binomial metric	Continuous metric w/censoring
40%	44	24
30%	74	38
20%	166	98

40-50% reduction in test size

If you do NOT need to convert back to a Probability (percentile) and can instead size test to detect difference in quantile (median time), then benefit is even greater!

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

How to Calculate Power

- **No closed form equation to determine in this case**
- **Standard method when no closed-form exists is to conduct a Monte Carlo**
- **Method:**
  - Establish the parameters (μ and σ) under the null hypothesis (e.g., $P_{detect} \leq 0.50$)
  - Establish the parameter to be tested (μ in this case) under the alternate hypothesis
    - » Assume some effect size of interest for probability-to-detect; this equates to a shift in μ
  - Simulate data under the alternate hypothesis
    - » For times that occur beyond the nominal event duration (e.g., 6-hour on-station time), the censor value is set to "1."
  - Conduct the analysis on the simulated dataset
    - » i.e., MLE determines fitted values of μ and σ
  - Determine the standard errors (or confidence intervals) for the parameters (and P_{detect}). Based on the standard errors and the selected alpha (1 – confidence) value chosen, determine if the fitted P_{detect} value is statistically different than the null hypothesis P_{detect} value
    - » If so, it's a "correct rejection" of the null
  - Repeat the above steps 10,000 times.
  - Power equals the fraction of correct rejections
- **Note that Type 1 Error does not necessarily equal the alpha value you chose! Must check when doing power calculations….**
  - For censored data analyses, type 1 error (chance of wrongly rejecting null when it's true) is higher than alpha when:
    - » Small data sets
    - » High censoring

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Conclusions

- **Many binary metrics can be recast using a continuous metrics**
  - Care is needed, does not always work, but…
  - Cost saving potential is too great not to consider it!

- **With Censored-data analysis methods, we retain the binary information (non-detects), but gain the benefits of using a continuous metric**
  - Better information for the user
  - Maintains a link to the “Probability of…” requirements

- **Converting to the censored-continuous metric maximizes test efficiency**
  - As much as 50% reduction in test costs for near identical results in percentile estimates
  - Benefit is greatest when the goal is to identify significant factors (characterize performance)

Approved for public release; distribution is unlimited.

Approved for public release; distribution is unlimited.

Example R Code

```
so = Surv(d$ScaledDT, !d$ND, type = 'right')
f = survreg(
  so ~ APB + Rec.Factor + Type + Array + Noise + Type*Noise + Type*Array + Noise*Array + Type*Noise*Array,
  dist = 'lognormal',
  data = d)
summary(f)


predict(f, data.frame(Type = 'SSK', Array = 'A', APB = '9', Noise = 'Loud', Rec.Factor = 6.2))
```

Approved for public release; distribution is unlimited.