



# Plotting Likelihood-Ratio-Based Confidence Regions for Two-Parameter Univariate Probability Models

Christopher Weld, Andrew Loh & Lawrence Leemis

To cite this article: Christopher Weld, Andrew Loh & Lawrence Leemis (2020) Plotting Likelihood-Ratio-Based Confidence Regions for Two-Parameter Univariate Probability Models, The American Statistician, 74:2, 156-168, DOI: [10.1080/00031305.2018.1564696](https://doi.org/10.1080/00031305.2018.1564696)

To link to this article: <https://doi.org/10.1080/00031305.2018.1564696>



View supplementary material [↗](#)



Accepted author version posted online: 07 Feb 2019.  
Published online: 24 May 2019.



Submit your article to this journal [↗](#)



Article views: 212



View related articles [↗](#)



View Crossmark data [↗](#)



# Plotting Likelihood-Ratio-Based Confidence Regions for Two-Parameter Univariate Probability Models

Christopher Weld<sup>a</sup>, Andrew Loh<sup>b</sup>, and Lawrence Leemis<sup>b</sup>

<sup>a</sup>Department of Applied Science, William & Mary, Williamsburg, VA; <sup>b</sup>Department of Mathematics, William & Mary, Williamsburg, VA

## ABSTRACT

Plotting two-parameter confidence regions is nontrivial. Numerical methods often rely on a computationally expensive grid-like exploration of the parameter space. A recent advance reduces the two-dimensional problem to many one-dimensional problems employing a trigonometric transformation that assigns an angle  $\phi$  from the maximum likelihood estimator, and an unknown radial distance to its confidence region boundary. This paradigm shift can improve computational runtime by orders of magnitude, but it is not robust. Specifically, parameters differing greatly in magnitude and/or challenging nonconvex confidence region shapes make the plot susceptible to inefficiencies and/or inaccuracies. This article improves the technique by (i) keeping confidence region boundary searches in the parameter space, (ii) selectively targeting confidence region boundary points in lieu of uniformly spaced  $\phi$  angles from the maximum likelihood estimator and (iii) enabling access to regions otherwise unreachable due to multiple roots for select  $\phi$  angles. Two heuristics are given for  $\phi$  selection: an elliptic-inspired angle selection heuristic and an intelligent smoothing search heuristic. Finally, a jump-center heuristic permits plotting otherwise inaccessible multiroot regions. This article develops these heuristics for two-parameter likelihood-ratio-based confidence regions associated with univariate probability distributions, and introduces the R `conf` package, which automates the process and is publicly available via CRAN.

## ARTICLE HISTORY

Received March 2018  
Accepted November 2018

## KEYWORDS

Graphical methods;  
Numerical optimization;  
Parameter estimation.

## 1. Introduction

Confidence regions provide a simultaneous measure of the precision of parameter estimates (Cox and Oakes 1984, pp. 42–43). They account for dependence in the parameter estimates, and therefore represent their probabilistic relationships better than an assembly of individual confidence intervals. A grid-type search is an inefficient way to plot a confidence region (Meeker and Escobar 1995). Jaeger (2016) highlighted the computational advantages of an alternative likelihood-ratio-based radial profile technique, and demonstrates that runtime improvement can measure orders of magnitude faster. For a two-parameter distribution, this algorithm effectively reduces a two-dimensional problem to a one-dimensional problem. The techniques developed here augment Jaeger's work by (i) determining upper bounds on the search radius to account for locations where the log-likelihood function is undefined, (ii) optimizing the angles for the one-dimensional search problems in order to produce a smooth boundary for the confidence region and (iii) allowing the point-of-reference for the radial azimuth direction and distance to relocate from the MLE in order to reach confidence region areas otherwise inaccessible to the algorithm. This article addresses these heuristics for likelihood-ratio-based confidence regions for two-parameter univariate probability distributions.

After a brief introduction to the radial profile log-likelihood ratio technique in Section 2 and an illustration of parameter space restrictions in Section 3, an example given in

Section 4 highlights confidence region plot challenges. This article then introduces two heuristics improving its implementation in Sections 5 and 6. Both heuristic techniques aim to improve the spacing of confidence region boundary plot points produced through modifications to the method's  $\phi$  values—a radial azimuth parameter defined in Section 2. The first improving technique finds points whose spacing is approximately equidistant along the perimeter of the confidence region boundary. The second improving technique uses a progressive search heuristic that weights regions of rapid change with more points. Section 7 addresses repairs for radially inaccessible regions, and Section 8 provides a practical example of such a circumstance. Section 9 introduces the open-source R package `conf` which automates these processes. Finally, Section 10 contains a discussion of future work and Section 11 provides concluding remarks.

## 2. Radial Profile Log-Likelihood Ratio

This section briefly summarizes the radial profile log-likelihood ratio technique for plotting confidence regions summarized by Jaeger (2016).

Let  $\theta$  be a vector of  $p$  unknown parameters associated with a univariate probability distribution. Let  $L(\theta)$  be the likelihood function. Let  $\hat{\theta}$  be the corresponding vector of maximum likelihood estimates (MLEs). A confidence region for  $\theta$  at significance level  $\alpha$  is determined using an asymptotic result

associated with the likelihood ratio test statistic

$$-2 \left[ \log L(\theta) - \log L(\hat{\theta}) \right] \xrightarrow{D} \chi^2(p),$$

where  $\log$  is the natural logarithm. The boundary of a  $100(1 - \alpha)\%$  confidence region is all  $\theta$  values satisfying

$$-2 \left[ \log L(\theta) - \log L(\hat{\theta}) \right] = \chi_{p,\alpha}^2, \tag{1}$$

where the second subscript is associated with the right-hand tail probability. Calculating the boundary of this confidence region is computationally expensive using a progressive search or grid-exploration technique.

Now consider the case of  $p = 2$  unknown parameters,  $\theta_1$  and  $\theta_2$ , with associated MLEs  $\hat{\theta}_1$  and  $\hat{\theta}_2$ . This will be the case considered for the rest of this article. The radial profile log-likelihood technique (Jaeger 2016) pieces together the boundary of the confidence region by identifying boundary points on various cross-sections of the log-likelihood function. A vector of unique angles  $\phi$  from its MLE define the cross-sections, and the vector  $r$  represents their respective radial distances from the MLE to the boundary of the log-likelihood function. Adopting these modifications, Equation (1) becomes

$$-2 \left[ \log L(\hat{\theta}_1 + r \cos \phi, \hat{\theta}_2 + r \sin \phi) - \log L(\hat{\theta}_1, \hat{\theta}_2) \right] = \chi_{2,\alpha}^2. \tag{2}$$

By exploiting the asymptotically  $\chi^2(2)$  distribution of the likelihood ratio statistic, solving for  $r$  in Equation (2) identifies points on the boundary of its approximate  $100(1 - \alpha)\%$  confidence region. When taken in aggregate over the domain  $\phi \in [0, 2\pi)$ , these points delineate the confidence region of interest.

### 3. Parameter Space Restrictions

The radial distance,  $r$ , from the MLE to its confidence region boundary for given  $\phi$  angles can be calculated using Equation (2). For most two-parameter univariate probability distributions this has no closed-form solution, and requires numerical methods. Those solution methods become complicated, however, when the parameter space is restricted. For instance, implementing a bisection type algorithm—as available in the R `uniroot` function—requires lower and upper limits to bracket the solution. Naively specifying arbitrarily large upper limits for  $r$  can extend evaluation of the log-likelihood function out of the parameter space, thereby terminating the algorithm without solution. The upper limits on  $r$  must account for this possibility.

The parameter space for the distribution parameters,  $\theta$ , determines the feasible regions for  $r$ . Two-parameter distributions such as the gamma, inverse Gaussian, log logistic, and Weibull distributions have the first quadrant as a parameter space. Alternate and/or unique support constraints are dealt with accordingly. For example, a triangular( $0, \theta_1, \theta_2$ ) distribution would have its first quadrant feasible region restricted to the area where  $0 < \theta_1 < \theta_2$ . In another example, if one parameter can assume any real value and the second is nonnegative—such as the normal or log normal distributions—then the parameter space consists of the first and second quadrants (assuming the vertical axis represents the nonnegative parameter).

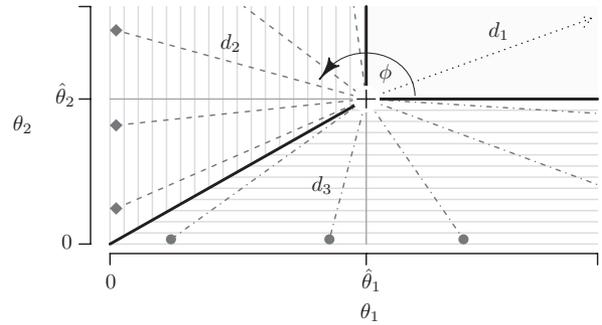


Figure 1. Log-likelihood feasible region constraints for radial distances,  $r$ , from the MLE given  $\theta_1 > 0, \theta_2 > 0$ .

Figure 1 illustrates constraints for the case when  $\theta_1 > 0, \theta_2 > 0$ , and  $r$  is restricted to the first quadrant. It is broken into three sections—each MLE dependent—separated by thicker solid lines that radiate from the MLE denoted by + at  $(\hat{\theta}_1, \hat{\theta}_2)$ . Dotted lines within this figure illustrate several  $\phi$  values for assessment, some of which have a  $d_i$  annotation for reference here. The section containing  $d_1$  associated with  $0 \leq \phi \leq \pi/2$  has no upper bound. Any length  $d_1$  remains feasible, therefore, an arbitrarily large upper limit is sufficient. Care is taken within the remaining two regions, however, to capture feasible possibilities without extending to its infeasible region. The symbols  $\blacklozenge$  and  $\bullet$  show desired upper bounds to restrict the search for  $r$ , lying arbitrarily close to each axis but still within the first quadrant. Trigonometric relationships reveal these respective upper bounds, each representative of all  $\phi$  within their section, as

$$d_2 = a \cdot \left( \frac{-\hat{\theta}_1}{\cos \phi} \right) \quad \pi/2 < \phi < \pi + \arctan(\hat{\theta}_2/\hat{\theta}_1) \quad \text{and}$$

$$d_3 = a \cdot \left( \frac{-\hat{\theta}_2}{\sin \phi} \right) \quad \pi + \arctan(\hat{\theta}_2/\hat{\theta}_1) \leq \phi < 2\pi,$$

where  $a \uparrow 1$  to keep its result within the first quadrant.

### 4. Example of Implementation Challenges

The  $n = 23$  deep-groove ball bearing failure times, given by Lieblein and Zelen (1956) in millions of revolutions, will serve as our example throughout this article:

- 17.88, 28.92, 33.00, 41.52, 42.12, 45.60,
- 48.48, 51.84, 51.96, 54.12, 55.56, 67.80,
- 68.64, 68.64, 68.88, 84.12, 93.12, 98.64,
- 105.12, 105.84, 127.92, 128.04, 173.40.

A Weibull distribution is chosen to model the data. It has a survivor function

$$S(x) = e^{-(\lambda x)^\kappa} \quad x > 0,$$

with positive shape parameter  $\kappa$  and positive scale parameter  $\lambda$ . The associated log-likelihood function is

$$\log L(\lambda, \kappa) = n \log \kappa + n \kappa \log \lambda + (\kappa - 1) \sum_{i=1}^n \log x_i - \lambda^\kappa \sum_{i=1}^n x_i^\kappa,$$

where  $x_1, x_2, \dots, x_n$  denote the data values, with corresponding MLEs  $\hat{\lambda} = 0.0122$ , and  $\hat{\kappa} = 2.102$ .

Let  $m$  be the length of  $\phi$  (number of angles) in the radial profile log-likelihood ratio technique. Assessing  $m = 360$  and  $m = 3600$  equally-spaced angles from the MLE,  $\phi \in \{0, 2\pi \cdot \frac{1}{m}, 2\pi \cdot \frac{2}{m}, 2\pi \cdot \frac{3}{m}, \dots, 2\pi \cdot \frac{m-1}{m}\}$ , the radial profile log-likelihood ratio technique yields the plots shown in Figure 2. Each plot point in Figure 2 is found by solving equation (2) for  $r$  using the `uniroot` function in R for the ball bearing failure times. The fact that  $\kappa = 1$  falls outside of the confidence region indicates that there is statistical evidence that the population distribution is in the IFR (increasing failure rate) class. The confidence region supports the intuitive notion that the ball bearings are wearing out.

Sharp vertices and nonuniform point distributions in both graphs in Figure 2 are a cause for concern. Even in Figure 2 (right), with 3600 angles assessed (spaced every  $1/10^\circ$ ), the plot lacks sufficient smoothness on its right side. These issues are traceable to the vastly different magnitudes for  $\hat{\lambda}$  and  $\hat{\kappa}$ . Figure 3 illustrates how the ratio of these quantities influences the distribution of points along an elliptical boundary under eight equally spaced angles  $0 \leq \phi < 2\pi$  on a circle and on an ellipse associated with generic parameters  $\theta_1$  and  $\theta_2$ .

Equally spaced  $\phi$  angles become increasingly less effective as the relative axis scales get further apart. Some areas—nearer to the top and bottom of both Figures 2 and 3—plot more points than necessary, while others areas—nearer to the left and right of both Figures 2 and 3—suffer from insufficient smoothness. In Figure 2, the height and width scales differ over two orders of magnitude; therefore, spacing implications are much more severe than what Figure 3 demonstrates. Targeted modifications to  $\phi$  angles assessed within the radial profile log-likelihood plotting technique must address these deficiencies for it to remain effective, and two such techniques are given in the next two sections.

### 5. Elliptically Oriented Points

Identifying equidistant points along the confidence region boundary is an intuitive approach to improve upon the results in Figure 2. This, however, implies having a priori knowledge of its shape and size, which are both unknown. Nonetheless, by estimating its relative size using an ellipse to account for

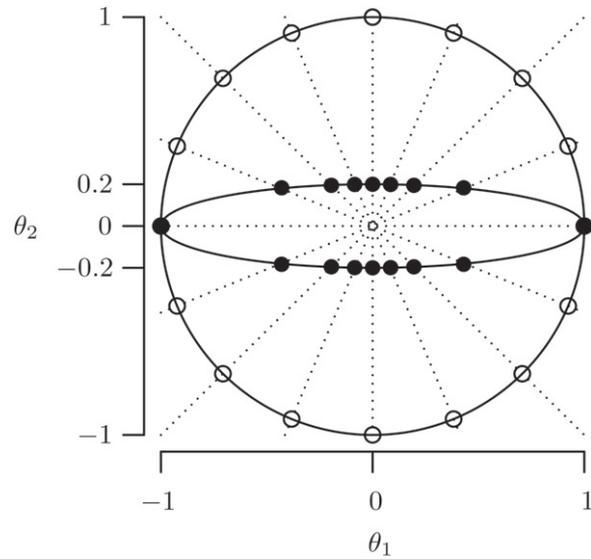


Figure 3. Point spacing comparison between 1:1 and 5:1 width-to-height ratio elliptical plots given 16 uniformly distributed angles for  $0 \leq \phi < 2\pi$ .

significant differences in the magnitudes of the parameters, results improve significantly.

An ellipse is chosen to approximate the confidence region shape because a  $p$ -dimensional confidence region converges to a  $p$ -dimensional ellipsoid as  $n \rightarrow \infty$ . This result is implicit in the asymptotic result

$$\sqrt{n}(\hat{\theta} - \theta) \xrightarrow{D} N(0, I^{-1}(\theta)),$$

where  $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2, \dots, \hat{\theta}_p)'$  denotes the vector of MLEs of the unknown parameters  $\theta = (\theta_1, \theta_2, \dots, \theta_p)'$ , and  $I(\theta)$  is the information matrix associated with the random sample  $x_1, x_2, \dots, x_n$  in estimating  $\theta$ .

We choose the Steiner generation of a nondegenerate conic section to construct an ellipse with points that are themselves approximately equally spaced along the ellipse circumference. This algorithm, also known as the parallelogram method, is a result of the Theorem of Steiner (Meserve 1983, p. 65) and is given in greater detail next.

Figure 4 illustrates the parallelogram method with generic parameters  $\theta_1$  and  $\theta_2$  having respective major and minor axis lengths of  $2a$  and  $2b$ . This example assumes the ellipse major

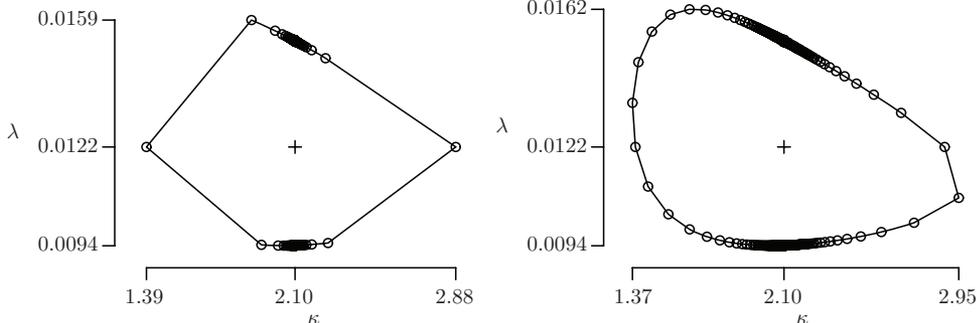


Figure 2. Confidence regions for  $\kappa$  and  $\lambda$  for the ball bearing failure times fitted to the Weibull distribution for  $\alpha = 0.05$ , and  $0 \leq \phi < 2\pi$  uniformly distributed using  $m = 360$  (left) and  $m = 3600$  (right) angles.

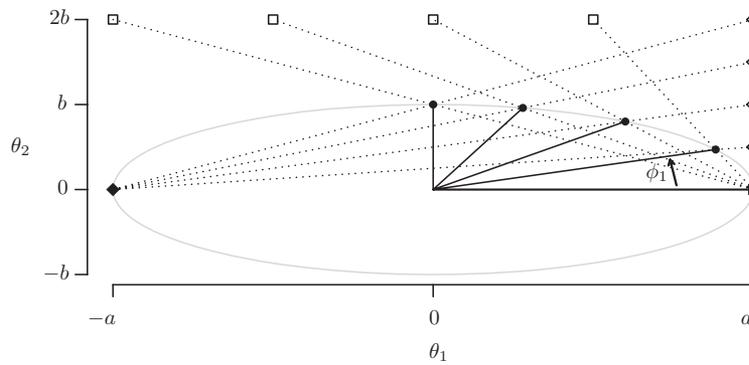


Figure 4. Parallelogram method for points on an ellipse with major and minor axes lengths of  $2a$  and  $2b$ .

axis—having  $\blacklozenge$  and  $\blacksquare$  endpoint symbols—is parallel to the horizontal axis. To identify  $m = 4n$  ellipse points, place  $n$  points ( $\blacklozenge$ ) equally spaced above a major axis endpoint ( $\blacksquare$ ) from  $(0, 2b]$  and connect them with line segments to the opposite major axis endpoint ( $\blacklozenge$ ). Next, place  $n$  points ( $\square$ ) above and parallel to the major axis, offset at a distance equal to the minor axis ( $2b$ ), and spaced equidistant from  $[-a, a)$ . Connect them ( $\square$ ) to the other major axis endpoint ( $\blacksquare$ ). The intersection of these respective line segments (sequencing  $\square$  left-to-right and  $\blacklozenge$  top-to-bottom) identifies  $n$  points ( $\bullet$ ) along the ellipse. Ellipses are symmetric about their major and minor axes, therefore the remaining  $3n$  points are easily attainable. The points are approximately equidistant.

A shortfall remains: appropriate ellipse dimensions are unknown. Considering the parameters of interest are the set of radial angles  $\phi$ , only the relative size of the ellipse’s major and minor axes (its elongation or eccentricity) is relevant. The  $\phi$  values depend only on this elongation and  $m$ . An approximation is sufficient; precision is not critical to produce a series of angles that adequately distribute along the confidence region boundary. The  $\phi$  angles are, after all, approximations themselves of where true equidistant points along the confidence region boundary might lie.

The ratio of asymptotic standard errors of its MLEs identifies a reasonable approximation of the ellipse elongation. It is found by first calculating the  $2 \times 2$  Fisher information matrix for the Weibull distribution using the partial derivatives

$$\begin{aligned} \frac{-\partial^2 \log L(\lambda, \kappa)}{\partial \lambda^2} &= \frac{\kappa n}{\lambda^2} + \kappa(\kappa - 1) \lambda^{\kappa-2} \sum_{i=1}^n x_i^\kappa, \\ \frac{-\partial^2 \log L(\lambda, \kappa)}{\partial \lambda \partial \kappa} &= -\frac{n}{\lambda} + \lambda^{\kappa-1} \\ &\quad \times \left[ \kappa \sum_{i=1}^n x_i^\kappa \log x_i + (1 + \kappa \log \lambda) \sum_{i=1}^n x_i^\kappa \right], \\ \frac{-\partial^2 \log L(\lambda, \kappa)}{\partial \kappa^2} &= \frac{n}{\kappa^2} + \sum_{i=1}^n (\lambda x_i)^\kappa (\log \lambda x_i)^2. \end{aligned}$$

Although the expected values of these partial derivatives have no closed-form solution, the observed information matrix is attainable using  $\hat{\lambda}$  and  $\hat{\kappa}$ . For the ball bearing failure times, the observed information matrix is

$$O(\hat{\lambda}, \hat{\kappa}) = \begin{bmatrix} 681,000 & 875 \\ 875 & 10.4 \end{bmatrix}.$$

The inverse of the observed information matrix,

$$O^{-1}(\hat{\lambda}, \hat{\kappa}) = \begin{bmatrix} 0.00000165 & -0.000139 \\ -0.000139 & 0.108 \end{bmatrix},$$

identifies the estimated variance of  $\hat{\lambda}$  and  $\hat{\kappa}$  in its diagonal. The ratio of MLE asymptotic standard errors is therefore  $s_{\hat{\kappa}}/s_{\hat{\lambda}} = \sqrt{0.108}/\sqrt{0.00000165} = 256$ .

The set of  $\phi$  angles corresponding to the approximately equidistant ellipse points are now ready for use in the radial log-likelihood plotting technique. These  $\phi$  angles differ greatly from equally spaced angles. As the width-to-height ratio increases,  $\phi$  angles concentrate nearer  $0^\circ$  and  $180^\circ$ , as indicative of the trend shown in Figure 4. This effect is even more pronounced for the ball bearing dataset, whose elongation is much more severe.

Results using this technique on the ball bearing data set are shown in Figure 5 for  $m = 100$ . Appendix A contains a description of Algorithm 1, which implements this elliptically oriented points approach.

Using the parallelogram method to identify  $\phi$  angles results in a more uniform distribution of points along the confidence region’s boundary, significantly improving both plot resolution and its associated computational cost. Using R, Figure 5 took just 0.070 sec runtime and gave better results than both Figure 2 (left) using  $m = 360$  points at 0.285 sec and Figure 2 (right) using  $m = 3600$  points at 2.812 sec.

Although a significant improvement over using equally spaced  $\phi$  angles, this technique has vulnerabilities. A severe bend between points may result in visually striking and

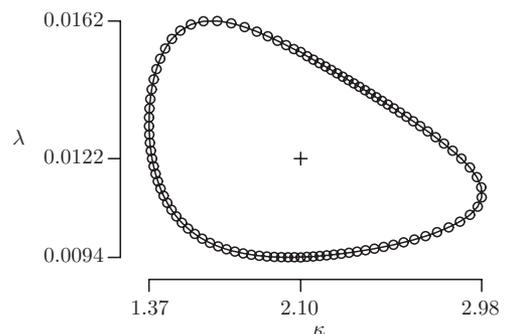


Figure 5. Confidence region for  $\kappa$  and  $\lambda$  for the ball bearing failure times fitted to the Weibull distribution using the elliptically oriented heuristic described in Algorithm 1 with  $\alpha = 0.05$ ,  $m = 100$ , and a major to minor axes ratio of  $s_{\hat{\kappa}}/s_{\hat{\lambda}} = 256$ .

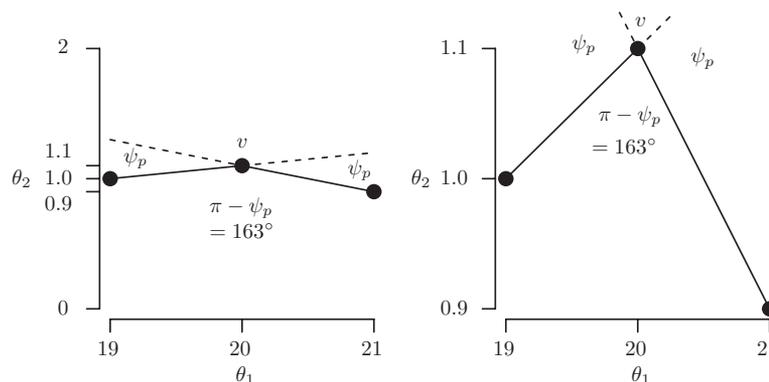
misleading vertex angles along its boundary. An example of this is given in Section 8. Although the computational ease and efficiency of the elliptically oriented algorithm for identifying confidence region boundary points make it a worthwhile option to consider, portions of confidence region boundaries with high curvature may warrant a more robust approach. The next section addresses such cases with an alternate strategy.

## 6. Smoothing Boundary Search Heuristic

In their analysis of confidence curves for nonlinear regression, Cook and Weisberg (1990) developed a dynamic step size approach after recognizing more plot points were necessary to adequately model regions where the likelihood function changes rapidly. An analogous logic motivates this heuristic, which strategically identifies points along the confidence region boundary that result in a smooth and accurate plot.

Accuracy is assumed for points evaluated along the confidence region boundary by a numerical solver. Linear connections assumed between those points, however, are an approximation of the boundary's true shape. Given a sufficient number of points, those linear approximations are indistinguishable from the true region. The objective of the smoothing search algorithm is to select more points along the confidence region boundary where its curvature is large, in contrast to areas with smaller curvature where fewer points are adequate to approximate its true shape. Adequate plot resolution is loosely quantified as a sequence of adjoining line segments such that vertex angles are indistinguishable; the boundary appears to be a smooth curve. To quantify the smoothness with greater specificity, a maximum tolerable vertex angle is assigned.

Creating a plot that conforms to a maximum tolerable vertex angle constraint is nontrivial. This is because the apparent plot angle (the angle *as it appears* in the plot) differs from its actual angle (the calculable angle using trigonometry) when its respective horizontal and vertical axes limits cover ranges disproportionate to their relative plot space (the plot width and height). These impacts are illustrated in Figure 6 for its vertex angle at point  $v$ . Its plots show identical vertex angles—they have identical points—with different  $\theta_2$  vertical axis limits. Within a square plot area (plot width and height are equal) nonequal axes limit ranges will distort angles, as evident in Figure 6



**Figure 6.** Identical plots using different vertical axes limit ranges. Identical horizontal and vertical axes limit ranges (left) in a square plot area result in an actual angle equivalent to its apparent angle, whereas differing axes limit ranges (right) distort its resulting apparent angle.

(right). Under such warping dynamics, an acceptable angle threshold loses meaning. An in-tolerance angle may appear out-of-tolerance, and vice versa.

This complication is overcome using a consistent methodology to frame the confidence region, in conjunction with a transformation. We define the plot area using a minimum bounding rectangle, as seen in Figure 6 (right), where the axes limits adjust to accommodate no more than their minimum and maximum support values. Strictly imposing this standard gives our methodology a consistent framework to operate within. The  $n$  points defining the confidence region boundary, given by the coordinate pairs  $(\theta_{1,i}, \theta_{2,i})$  for  $i = 1, 2, \dots, n$ , are then transformed to  $(\theta_{1,i}, \theta'_{2,i})$  according to

$$\theta'_{2,i} = s \cdot \theta_{2,i} \quad \text{for } i = 1, 2, \dots, n,$$

$$\text{where } s = \frac{\max\{\theta_{1,1}, \theta_{1,2}, \dots, \theta_{1,n}\} - \min\{\theta_{1,1}, \theta_{1,2}, \dots, \theta_{1,n}\}}{\max\{\theta_{2,1}, \theta_{2,2}, \dots, \theta_{2,n}\} - \min\{\theta_{2,1}, \theta_{2,2}, \dots, \theta_{2,n}\}}.$$

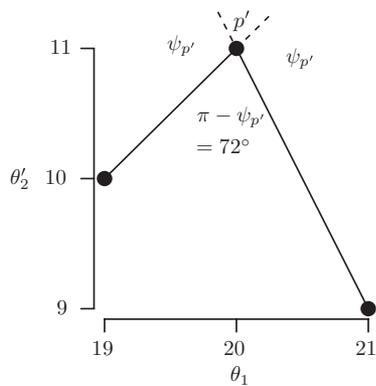
Within a square plot area, this transformation yields identical apparent and actual angles; the distortion effects of Figure 6 (right) are accounted for. The algebraic manipulation below, in which  $\theta_j$  represents  $\theta_{j,1}, \theta_{j,2}, \dots, \theta_{j,n}$  for each  $j = 1, 2$ , confirms this by demonstrating the equivalence of the transformed vertical axis limits range with the horizontal axis limits range.

$$\begin{aligned} \max\{\theta'_2\} - \min\{\theta'_2\} &= s \cdot \max\{\theta_2\} - s \cdot \min\{\theta_2\} \\ &= s \cdot (\max\{\theta_2\} - \min\{\theta_2\}) \\ &= \frac{\max\{\theta_1\} - \min\{\theta_1\}}{\max\{\theta_2\} - \min\{\theta_2\}} \\ &\quad \cdot (\max\{\theta_2\} - \min\{\theta_2\}) \\ &= \max\{\theta_1\} - \min\{\theta_1\}. \end{aligned}$$

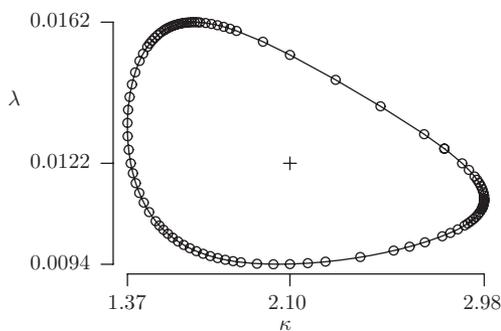
Figure 7 shows this transformation, correctly identifying the apparent angle in Figure 6 (right) as  $72^\circ$ .

This transformation enables the design of Algorithm 2, provided in Appendix B. In it, a maximum tolerable vertex angle  $\psi_{\text{tol}}$ —the apparent angle as seen in a minimum bounding boxplot—is assigned. Smaller values of  $\psi_{\text{tol}}$  result in smoother boundaries. By first transforming the plot, apparent angles are calculable, and those not within tolerance are augmented with additional confidence region boundary points accordingly.

Figure 8 shows the result of applying Algorithm 2 to the ball bearing failure times fit to the Weibull distribution. Boundary regions with greatest curvature—its top-left and extreme-right—accordingly have the greatest density of points. Figure 8



**Figure 7.** Figure 6 (right) under the transformation  $\theta'_2 = s \cdot \theta_2 = 10\theta_2$  yields equivalent axes ranges, therefore apparent and actual vertex angles match given a square plot area is in use.



**Figure 8.** Confidence region for  $\kappa$  and  $\lambda$  for the ball bearing failure times fitted to the Weibull distribution using the smoothing search heuristic described in Algorithm 2 with  $\alpha = 0.05$  and  $\psi_{\text{tol}} = 5^\circ$  maximum vertex angle tolerance ( $m = 102$  points result).

uses points along the confidence region boundary more efficiently than Figure 2. It achieves greater resolution of high curvature areas (located on its far-right and upper-left regions), and sacrifices little (seemingly identical results) by locating fewer points in relatively straight sections of the boundary. Although Algorithm 2 is elegant and uses confidence region boundary points efficiently, its iterative nature comes at a computational cost, taking 0.130 sec to run, or 1.9 times the elliptical method runtime. This trade-off is reasonable considering its improvement in the high-curvature area resolution, and the guaranteed nature of its results ( $\psi_{\text{tol}}$  constraint is active, whereas the elliptical method provides no guarantee that an unwanted “sharp” angle will not appear).

## 7. Repairs for Radially Inaccessible Regions

A second implementation challenge exists in addition to Section 4 issues of scale. Nonconvex log-likelihood function confidence region contours can result in confidence region shapes with area(s) inaccessible via the radial profile technique. These inaccessible regions are the result of multiple confidence region boundary points at select radial angles from the MLE, and become increasingly problematic for smaller sample sizes and/or smaller significance levels. The elliptical approach cannot plot these areas, and the smoothing-search algorithm terminates without satisfying its maximum degree tolerance.

This section improves the heuristic described in Section 6 by accessing these otherwise unreachable areas. The complete details for this approach given in general terms are available in Appendix C, Algorithm 3, and examples of its execution are given in Sections 8 and 9.

Figure 9 illustrates two inaccessible region scenarios, both with shapes having multiple roots for a subset of  $\phi$ . The counter-clockwise or clockwise direction-of-approach to its inaccessible region is what differentiates the scenarios. Its  $\bullet$  plot points share a near-identical radial angle from its MLE (+), shown as a dashed line. The line segment between the  $\bullet$  points marks the edge of its accessible and inaccessible (shaded) confidence regions. For reference, its location relative to the MLE is given quadrant labels I, II, III, and IV. Quadrant III techniques are transferable with trivial geometric and trigonometric manipulation to all other quadrants.

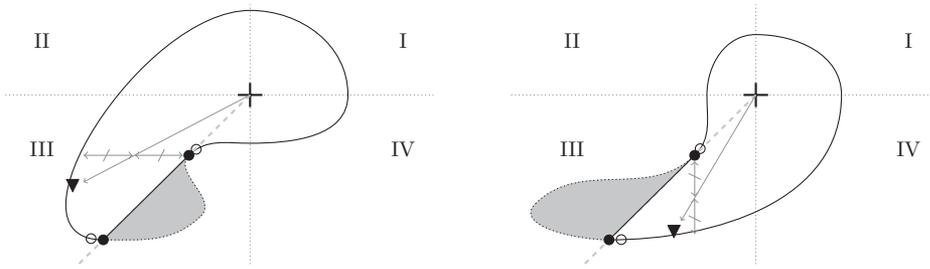
To access the uncharted confidence region area, an alternate “centerpoint” (not the MLE) is created within the existing confidence region perimeter, hereafter known as a *jump-center*. It is given by  $\blacktriangledown$  in Figure 9. Appropriately locating the jump-center requires identifying the respective orientation of the inaccessible region: a Figure 9 (left) or Figure 9 (right) scenario. This is done by comparing the  $y$ -coordinate values for points adjacent to the inaccessible region border, shown as  $\circ$  in Figure 9. For quadrant III (also shown in Figure 9), if the smaller (more highly negative)  $\circ$   $y$ -coordinate value corresponds to the smaller  $\phi$  angle, then it aligns with the Figure 9 (left) scenario, and Figure 9 (right) otherwise.

The jump-center is assigned as a slightly uphill point (relative to the three-dimensional log-likelihood surface) from the confidence region boundary for a given angle from the MLE. The angle chosen bisects the horizontal (Figure 9 left) or vertical (Figure 9 right) gap, where the *gap* represents the line segment within the confidence region “nearest” the inaccessible region and parallel to an axis. Although any angle crossing this gap is feasible, bisecting is a reasonable approach because it projects the jump-center into the vicinity of the uncharted region, however, not so near it that it might inadvertently locate at its nearer edge. From the jump-center, the radial profile method can identify points in the previously inaccessible region to combine with the original solution.

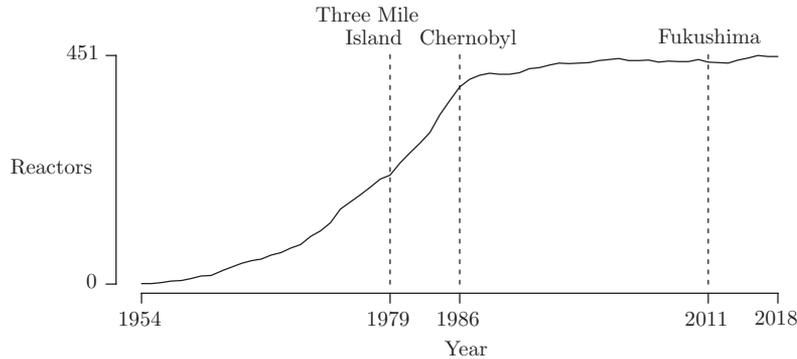
## 8. Jump-Center Repairs Example

Evans, Drew, and Leemis (2008) illustrated the case of small datasets with two applications: operating time between catastrophic space shuttle failures and between nuclear reactor meltdowns. This example will focus on nuclear reactor meltdowns, which currently number three: Three Mile Island in the USA in 1979, Chernobyl in the Ukraine in 1986, and Fukushima in Japan in 2011. Figure 10 shows the quantity of reactors worldwide in operation since the Russian Obninsk AM-1 reactor first came online in 1954, with data from the Nuclear Energy Institute (2018) and World Nuclear Association (2018). This analysis makes the simplifying assumption of reactor commissioning or decommissioning on the first day of the calendar year.

Total worldwide reactor operating time preceding each meltdown is found by integrating under Figure 10 curve, and occur



**Figure 9.** Hypothetical confidence region shapes with radially inaccessible (shaded) areas with respect to the MLE (+). Reference points ● and ○ help establish the jump-center location ▼ within the confidence region. Comparing ○ vertical values differentiates between scenarios (left and right).



**Figure 10.** Worldwide active nuclear reactors from 1954–2018.

at 1728, 3714, and 14,460 reactor-years, respectively. Let the random variable of interest be the time between core meltdowns. The  $n = 3$  samples are thus 1728, 1986, and 10,746 reactor-years.

Next, the likelihood-ratio based confidence region is sought using the radial approach for a gamma population distribution with probability density function

$$f(x) = \frac{1}{\Gamma(\kappa)\theta^\kappa} x^{\kappa-1} e^{-x/\theta} \quad x > 0,$$

with positive shape parameter  $\kappa$  and positive scale parameter  $\theta$ . The associated log-likelihood function is

$$\log L(\theta, \kappa) = -n \log \Gamma(\kappa) - n\kappa \log \theta + (\kappa - 1) \sum_{i=1}^n \log x_i - \sum_{i=1}^n \frac{x_i}{\theta}.$$

The MLEs for  $\theta$  and  $\kappa$  cannot be expressed in closed form, but can be calculated using numerical methods.

The plot heuristics from Sections 5 and 6 both have difficulty generating a confidence region for the  $n = 3$  times between meltdowns because of radially inaccessible regions. The elliptic approach (Figure 11 left) fails to adequately locate points in areas of relatively high curvature (top-left and bottom-right regions). The smoothing search approach (Figure 11 right) terminates at the maximum iteration tolerance without satisfying its maximum vertex degree constraint,  $\psi_{\max} \leq 5^\circ$ , where  $\theta = 50,900$ . It reveals a large gap between ● plot points sharing a near-identical angle from its MLE (+), shown as a dotted line.

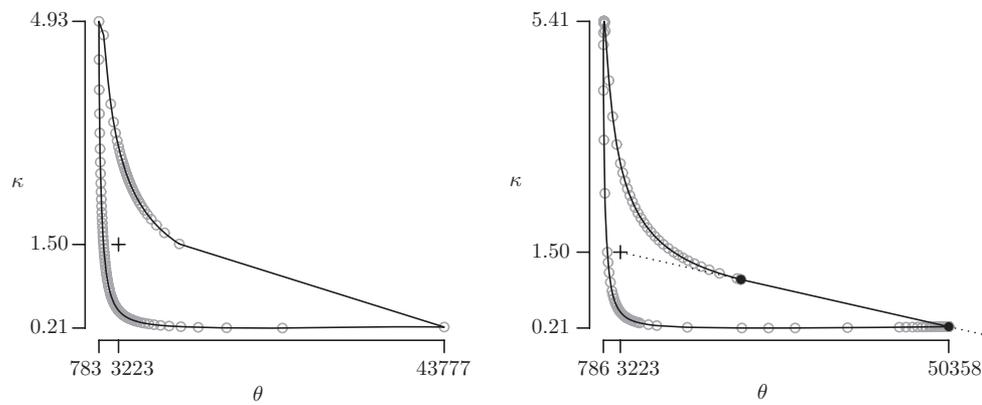
Figure 12 (left) illustrates the implementation of the heuristic for handling radially inaccessible regions from Section 7, including the jump-center location (▼) and the additional confidence region points it creates (▽). Figure 12 (right) shows the final form of the 90% confidence region.

Two comments conclude this section. First, Section 7 repairs also prove valuable with larger datasets when significance levels are small. For example, the Weibull confidence region for the  $n = 23$  ball bearing dataset from Section 4 requires repair when  $\alpha \leq 10^{-11}$ . Second, confidence regions can have multiple inaccessible regions requiring repair. An example of this nature is given in Section 9.

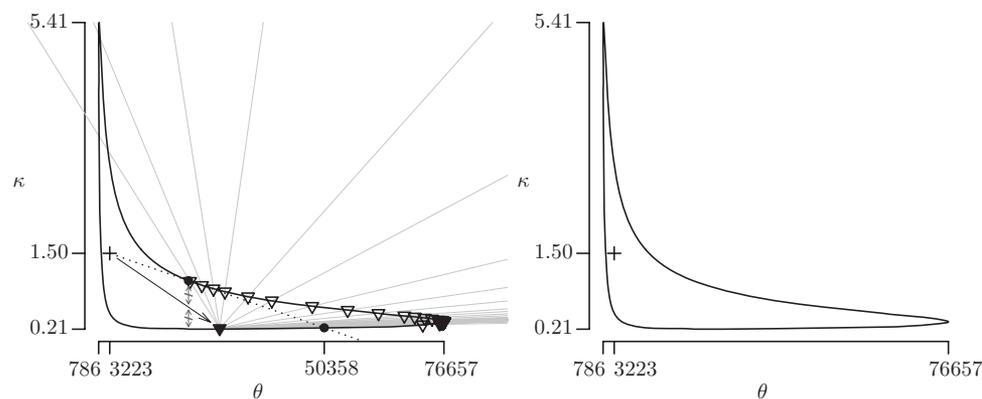
## 9. R Package `conf`

The `crplot` function within the R package `conf` automates confidence region plots, and is publicly available from the Comprehensive R Archive Network (Weld, Park, and Leemis 2018). It currently supports nine distributions: the Cauchy, gamma, inverse Gaussian, logistic, log-logistic, log-normal, normal, uniform, and Weibull distributions. This section describes the `crplot` required and optional arguments, and then provides examples of its syntax and output.

The required arguments for `crplot` are: data values (`dataset`), significance level (`alpha`), and distribution name (`dist`) using R suffixes: `cauchy`, `gamma`, `invgauss`, `logis`, `llogis`, `lnorm`, `norm`, `unif`, and `weibull`. The binary vector `cen` specifies if corresponding dataset values are right-censored (0) or observed (1, default). Algorithm 2 smoothing search heuristic is used by default with  $\psi_{\text{tol}} = 5^\circ$



**Figure 11.** Confidence region for  $\theta$  and  $\kappa$  for the  $n = 3$  times between meltdowns fitted to the gamma distribution for  $\alpha = 0.1$  using the elliptical heuristic algorithm from Section 5 with  $m = 100$  points (left) and the smoothing search heuristic from Section 6 naively applied (right).



**Figure 12.** Confidence region for  $\theta$  and  $\kappa$  for the number of reactor-years between nuclear meltdowns fitted to the gamma distribution using the jump-center repairs heuristic in Algorithm 3 to augment the smoothing search heuristic of Algorithm 2, shown with jump-center reference points (left) and in its final form (right) for  $\alpha = 0.1$ .

maximum degree tolerance (adjustable with the `maxdeg` argument). Repairs to radially inaccessible regions using Algorithm 3 are also invoked by default, and turned off using `repair = FALSE`. Its parameters enable customizing jump-center location via gap width shift percentage and its relative uphill location from the confidence region boundary (see Section 7 for descriptions), and are given default values `jumpshift = 0.5` and `jumpuphill = min(alpha, 0.01)`.

Algorithm 1 elliptic heuristic with  $m$  plot points is called with `heuristic = 0` and `ellipse_n = m`, where `ellipse_n` is a multiple of 4 and  $\geq 8$  to exploit computational efficiency associated with ellipse symmetry in its respective quadrants. Providing an `ellipse_n` value without specifying `heuristic = 0` combines Algorithms 1 and 2, first plotting points with the elliptically oriented heuristic and subsequently complementing them accordingly to meet constraints set by the smoothing search heuristic.

Plot points and the MLE location are hidden using `pts = FALSE` and `mlelab = FALSE`, respectively. Jump-center reference points, analogous to Figure 12 (left), are plot using `showjump = TRUE`. Significant figures for horizontal and vertical axes are specified using `sf = c(x, y)`, where  $x$  and  $y$  represent the respective values of the optional `digits` argu-

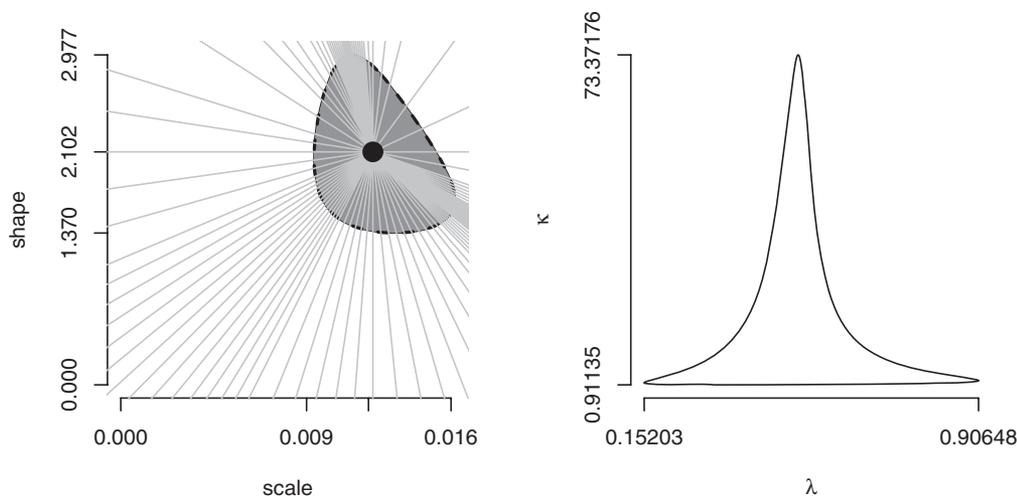
ment in the R function `round` as it pertains to those respective axes labels. Customization of margin size, main plot label, axes labels and orientation, and axes limits are also possible, and all conform to R base graphics syntax. The horizontal and vertical axes are switched using `xyswap = TRUE`. Axes limits are coerced to include the point  $(0, 0)$  with `origin = TRUE`.

Plot information— $p$  plot points with corresponding  $\phi$  angles, and MLE values  $\hat{\theta}_1, \hat{\theta}_2$ —are returned as components in an R list with `info = TRUE`. This permits additional analysis and/or plot customization. Setting `jumpinfo = TRUE` will augment the list with jump-center repair information (when applicable). Either information request can be combined with `showplot = FALSE` to hide plot results. These features together motivate additional `conf` coverage simulation capabilities. While those details are omitted here, its `coversim` function is capable of iterating confidence region trials using random (or user specified) datasets to assess actual coverage of true population parameters (or a user specified point of interest) at a given stated nominal coverage probability.

Table 1 illustrates the use of `crplot` syntax given above, assuming the R vector `ballbearing` contains the 23 data values given in Section 4. It includes both references from throughout this article, and additional examples shown in Figure 13.

**Table 1.** Example R syntax to use the `crplot` function in the `conf` package.

crplot syntax	reference
<code>crplot(ballbearing, 0.05, "weibull", heuristic = 0, ellipse_n = 100, sf = c(2, 4), ylas = 1)</code>	Figure 5
<code>crplot(ballbearing, 0.05, "weibull", sf = c(2, 4), ylas = 1)</code>	Figure 8
<code>crplot(c(1728, 1986, 10746), 0.1, "gamma", heuristic = 0, ellipse_n = 100, sf = c(0, 2), ylas = 1)</code>	Figure 11 (left)
<code>crplot(c(1728, 1986, 10746), 0.1, "gamma", repair = FALSE, sf = c(0, 2), ylas = 1)</code>	Figure 11 (right)
<code>crplot(c(1728, 1986, 10746), 0.1, "gamma", pts = FALSE, sf = c(0, 2), ylas = 1)</code>	Figure 12 (right)
<code>x &lt;- crplot(ballbearing, 0.05, "weibull", pts = FALSE, origin = TRUE, info = TRUE, sf = c(3, 3), xlab = "scale", ylab = "shape", xyswap = TRUE)</code>	
<code>polygon(x\$lambda, x\$kappa, col = "gray50", lty = 2, lwd = 2)</code>	
<code>segments(rep(x\$lambdahat, length(x\$phi)), rep(x\$kappahat, length(x\$phi)), x\$lambdahat + 1000 * x\$lambdahat * cos(x\$phi), x\$kappahat + 1000 * x\$lambdahat * sin(x\$phi), col = "gray")</code>	
<code>points(x\$lambdahat, x\$kappahat, pch = 19, cex = 2)</code>	Figure 13 (left)
<code>crplot(c(1.9, 2, 2.2), 0.01, "llogis", cen = c(1, 1, 0), pts = FALSE, mlelab = FALSE, sf = c(5, 5))</code>	Figure 13 (right)

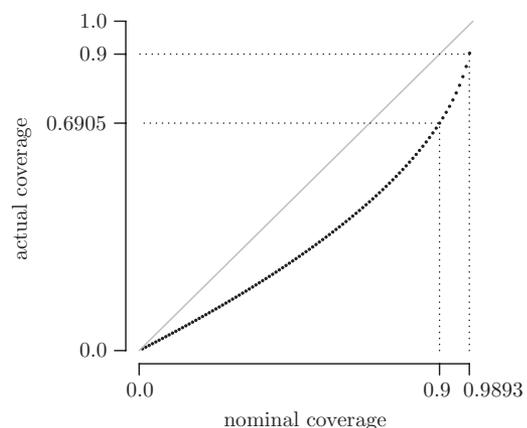
**Figure 13.** R output from Table 1 examples of confidence region plot customization using optional arguments. The left plot emphasizes the use `info = TRUE` to return plot info as a list enabling subsequent customization. The right plot demonstrates multiple Algorithm 3 repairs for a dataset with two observed and one right-censored value, and also hides the MLE location.

## 10. Future Work

Automating univariate confidence region plots enables coverage analysis to an extent unattainable in previous years. This insight is attributable to Jaeger's conclusion that the radial profile technique can be orders of magnitude faster than traditional grid search techniques (Jaeger 2016). Coverage analysis is relevant because it provides a signature of the underlying distribution characteristics, and can reveal new insights by way of its alternative perspective. For example, coverage analysis has led us to analytically derived actual coverage probabilities corresponding to stated nominal coverage for Weibull distributed samples of size  $n = 2$ . The mathematics are simple enough to compute the actual confidence region to infinite precision, where currently only estimates of its bias exists. Figure 14 illustrates those results, which were confirmed using 10,000 Monte Carlo simulation iterations for  $\alpha = 0.05, 0.1, \dots, 0.95$ . A confidence region with 90% nominal coverage for the Weibull parameters associated with  $n = 2$ , for example, has an actual coverage of only 69.05%. These smaller confidence regions result in a claim of more precision than is appropriate. Figure 14, which applies for all values of  $\lambda$  and  $\kappa$ , can be used to obtain an exact confidence region. If an actual coverage of 90% is desired, for example,

then constructing a confidence region associated with a 98.93% nominal confidence results in an exact 90% confidence region.

Additional potential resides in broadening the scope of implementation to allow for nuisance parameters, enabling

**Figure 14.** Precise actual confidence region coverage probabilities analytically derived given nominal coverage  $(1 - \alpha)$  for  $n = 2$  observations drawn from a  $\text{Weibull}(\lambda, \kappa)$  population for  $\alpha = 0.01, 0.02, \dots, 0.99$ .

three-parameter (three-dimensional) confidence region plots, and applying Algorithms 1–3 to circumstances other than univariate likelihood-ratio-based confidence regions. Performance using a Wald-based, or score-test-based confidence region can be pursued and compared. Alternatively, an analogous approach for bivariate data or regression parameters are worthwhile avenues to pursue. Doing so would allow access to computationally efficient confidence region plots, and empower similar simulations and coverage analysis.

## 11. Conclusion

The radial profile technique for plotting two-dimensional likelihood-ratio-based confidence regions is a tremendous tool, but should not be applied naïvely in all cases. It is important to also consider the parameter space feasible region, issues of scale, and inaccessible regions. Constraints on the parameter space are necessary to ensure numeric boundary point solutions are possible. Issues of scale arise when parameters differ greatly in magnitude. An inconsistent point distribution along the boundary of its confidence region results, an issue attributable to its uniformly distributed  $\phi$  values. Two heuristics for choosing improved  $\phi$  values devised here offer effective alternatives. The first identifies  $\phi$  angles coinciding with roughly equal point spacing along the circumference of an ellipse, which in turn leads to approximately equidistant points along the boundary of the confidence region. Its appeal includes ease of implementation and quick runtime. The second alternative is a smoothing search heuristic that allocates points along the confidence region boundary in an iterative fashion in the areas warranting the most attention. Its appeal includes efficient point distribution and guaranteed smoothing results—within an assigned vertex angle tolerance—at a reasonable computational cost. A final challenge arises when a single radial angle from the MLE crosses multiple confidence region boundary points—often if sample size and/or significance level are small—resulting in a radially inaccessible region. A jump-center heuristic allows access to these regions by relocating the point-of-reference for the radial approach away from the MLE. All heuristics within this article are automated with the R package `conf`, making likelihood-ratio based confidence regions for two-parameter univariate probability models and their subsequent coverage analysis readily accessible.

## Supplemental Material

The R package `conf` contains code for all confidence region plots and is accessible through the Comprehensive R Archive Network (CRAN). Data from the Nuclear Energy Institute (2018) and World Nuclear Association (2018) used to assemble Figure 10 are also included as supplemental material.

## Acknowledgments

The authors are grateful to the anonymous reviewers for their input which led to improvements in both article readability and the `crplot` function in the `conf` R package. The lead author is also grateful for partial funding provided by the Omar Bradley Research Fellowship in Mathematics.

## References

- Cook, D. R., and Weisberg, S. (1990), “Confidence Curves in Nonlinear Regression,” *Journal of the American Statistical Association*, 84, 410, 544–551. [160]
- Cox, D. R., and Oakes, D. (1984), *Analysis of Survival Data*, New York, Methuen. [156]
- Evans, D., Drew, J., and Leemis, L. (2008), “The Distribution of the Kolmogorov–Smirnov, Cramer–von Mises, and Anderson–Darling Test Statistics for Exponential Populations with Estimated Parameters,” *Communications in Statistics—Simulation and Computation*, 37, 1396–1421. [161]
- Jaeger, A. (2016), “Computation of Two- and Three-Dimensional Confidence Regions With the Likelihood Ratio,” *The American Statistician*, 70, 395–398. [156,157,164]
- Lieblein, J., and Zelen, M. (1956), “Statistical Investigation of the Fatigue Life of Deep-Groove Ball Bearings,” *Journal of Research of the National Bureau of Standards*, 57, 273–316. [157]
- Meeker, W. Q., and Escobar, L. A. (1995), “Teaching about Approximate Confidence Regions Based on Maximum Likelihood Estimation,” *The American Statistician*, 49, 48–53. [156]
- Meserve, B. E. (1983), *Fundamental Concepts of Geometry*, Mineola, NY: Dover Publications, Inc. [158]
- Nuclear Energy Institute (2018), *Nuclear Energy in the U.S. & Nuclear Energy in the World*, available at <https://nei.org/resources/statistics>. [161,165]
- Weld, C., Park, H., and Leemis, L. (2018), “`conf`: Visualization and Analysis of Statistical Measures of Confidence,” R package version 1.5, available at <https://CRAN.R-project.org/package=conf>. [162]
- World Nuclear Association (2018), *Decommissioning Nuclear Facilities*, available at <http://world-nuclear.org/information-library/nuclear-fuel-cycle/nuclear-wastes/decommissioning-nuclear-facilities.aspx>. [161,165]

## Appendix A. Elliptically Oriented Algorithm

The short description below summarizes Algorithm 1.

Lines 1–2 validate a usable  $m$  value is given; this constraint is necessary to take advantage of ellipse symmetry: calculations in one quadrant (with respect to the MLE) are transferable to the remaining three given  $m$  is divisible by four. Line 3 then stores the number of points in one quadrant. Line 4 initializes  $\phi$  as an empty set, and the MLEs are identified in Line 5. Line 6 stores values for the asymptotic standard error of the MLEs, which will dictate the ellipse eccentricity. Lines 7–13 then generate points along the circumference of an ellipse using the parallelogram method (reference Figure 4), also assembling the corresponding set of  $\phi$  angles to each point in Line 13 as it proceeds. With one quadrant of  $\phi$  angles determined, Line 14 exploits ellipse symmetry to populate the remaining three quadrants. Line 15 completes the set of  $\phi$  angles with cardinal directions at 0 and  $\pi$ . Points along the confidence region boundary corresponding to each  $\phi$  entry are finally identified in Line 17 using the radial profile log-likelihood ratio technique from Section 2, and returned as output to produce its plot.

**Algorithm 1:** Elliptically Oriented Algorithm for Identifying Confidence Region Boundary Points

---

```

input :  $x_1, x_2, \dots, x_n$  : data values
          $\alpha$  : confidence region (CR) significance level
          $m$  : number of points to plot (a multiple of four is required for this method)
          $\theta_1$  and  $\theta_2$  : unknown parameters from a parametric distribution
          $L(\theta_1, \theta_2)$  : likelihood function within the  $(\theta_1, \theta_2)$  parameter space
          $ase(\hat{\theta}_1, \hat{\theta}_2)$  : asymptotic standard errors of  $\hat{\theta}_1, \hat{\theta}_2$ 
          $h((A, B), (C, D))$  : function returning the point of intersection of line segments  $\overline{AB}$  and  $\overline{CD}$ 
          $g(\phi_i, \alpha)$  : function returning the CR boundary point associated with the angle  $\phi_i \in \langle \phi_1, \phi_2, \dots, \phi_m \rangle$  measured counterclockwise and centered at the MLE,  $0 \leq \phi < 2\pi$ 

output:  $p_i = (x_i, y_i)$  for  $i = 1, 2, \dots, m$  coordinate pairs distributed using the elliptically-oriented algorithm and enclosing a  $100(1 - \alpha)\%$  CR for  $\theta_1$  and  $\theta_2$ 
1 if  $m \% 4 \neq 0$  then
2   return{Error:  $m$  must be a multiple of four};           /* exit algorithm displaying error message */
3  $m_Q \leftarrow m/4$ ;                                       /* number of points to plot in one quadrant */
4  $\phi \leftarrow \{\}$ ;                                       /* initialize  $\phi$  as an empty set */
5  $(\hat{\theta}_1, \hat{\theta}_2) \leftarrow \operatorname{argmax}\{\log L(\theta_1, \theta_2)\}$ ; /* maximum log likelihood arguments are the MLEs */
6  $(s_{\hat{\theta}_1}, s_{\hat{\theta}_2}) \leftarrow ase(\hat{\theta}_1, \hat{\theta}_2)$ ;         /* identify MLE asymptotic standard errors */
7  $V_1 \leftarrow (s_{\hat{\theta}_1}, 0)$ ;                               /* point at the right end of the ellipse horizontal axis */
8  $V_2 \leftarrow (-s_{\hat{\theta}_1}, 0)$ ;                             /* point at the left end of the ellipse horizontal axis */
9 for  $i$  from 1 to  $m_Q$  do
10   $z_1 \leftarrow s_{\hat{\theta}_1} - 2s_{\hat{\theta}_1}i/m_Q$ ;                 /* x-coordinate of points to connect to  $V_1$  */
11   $z_2 \leftarrow s_{\hat{\theta}_2}i/m_Q$ ;                           /* y-coordinate of points to connect to  $V_2$  */
12   $(e_x, e_y) \leftarrow h((V_1, (z_1, s_{\hat{\theta}_2})), (V_2, (s_{\hat{\theta}_1}, z_2)))$ ; /*  $(x, y)$  coordinate of point on ellipse */
13   $\phi \leftarrow \{\phi, \arctan(e_y/e_x)\}$ ;                 /* augment  $\phi$  according to ellipse point  $(e_x, e_y)$  */
14  $\phi \leftarrow \{\phi, \pi - \phi, \pi + \phi, 2\pi - \phi\}$ ;   /* use ellipse symmetry to identify additional  $\phi$  angles */
15  $\phi \leftarrow \{\phi, 0, \pi\}$ ;                             /* augment  $\phi$  with 0 and  $\pi$  angles */
16 for  $i$  from 1 to  $m$  do
17    $(x_i, y_i) \leftarrow g(\phi_i, \alpha)$ ;                 /* identify CR points for each  $\phi_i$  */

```

---

**Appendix B. Smoothing Boundary Search Heuristic**

The short description below summarizes [Algorithm 2](#).

Line 1 initializes  $\phi$  values in each cardinal direction. Line 2 initializes a count variable and a variable to hold the maximum angle in the working plot,  $\psi_{\max}$ . The MLEs are identified in Line 3. A series of steps within the main `while` loop (Lines 4–35) then determine its confidence region boundary plot points, and if and where additional points are necessary. To do so, the radial log-likelihood function (described in [Section 2](#)) first determines confidence region points corresponding to each  $\phi$  value (Lines 7–9). Lines 10–15, then perform a transformation of the  $\theta_2$  values so that apparent  $\psi$  vertex angles are calculable, determined in Lines 17–21 using the law of cosines. Note that Line 16

preceding this calculation simplifies its execution by dictating the last point precedes the first, and the first point follows the last in its enclosed confidence region boundary. An analogous circumstance applies to Lines 23 and 24. Lines 26–34 conclude the `while` loop by adding points to the confidence region boundary in the vicinity of points where the  $\psi_{\text{tol}}$  constraint is not yet met. New points are identified using the angle through the midpoint of the existing adjacent points. This loop repeats, augmenting points and reevaluating vertex angles until  $\psi_{\text{tol}}$  is met at all confidence region vertexes, which concludes with a smooth confidence region for an appropriate choice of  $\psi_{\text{tol}}$ . If the maximum iteration tolerance ( $\text{count}_{\max}$ ) is met prior to satisfying  $\psi_{\text{tol}}$  then the program terminates with a warning and returns the working solution (indicative of inaccessible regions shown in [Section 7](#)).

**Algorithm 2:** Smoothing Search Heuristic for Identifying Confidence Region Boundary Points

```

input :  $x_1, x_2, \dots, x_n$  : data values
          $\alpha$  : confidence region (CR) significance level
          $\psi_{\text{tol}}$  : maximum angle tolerance between consecutive plot segments
         countmax : maximum iteration tolerance in algorithm before forced termination
          $\theta_1$  and  $\theta_2$  : unknown parameters from a parametric distribution
          $L(\theta_1, \theta_2)$  : likelihood function within the  $(\theta_1, \theta_2)$  parameter space
          $g(\phi_i, \alpha)$  : function returning CR boundary point associated with angle  $\phi_i \in \langle \phi_1, \phi_2, \dots, \phi_m \rangle$  measured counterclockwise and centered at the MLE,
          $0 \leq \phi < 2\pi$  corresponding to  $\alpha, \psi_{\text{tol}}$ 
          $d((x_i, y_i), (x_j, y_j))$  : function returning the length of the segment joining its respective points
output:  $p_i = (x_i, y_i)$  for  $i = 1, 2, \dots, m$  coordinate pairs enclosing a  $100(1 - \alpha)\%$  CR for  $\theta_1$  and  $\theta_2$  satisfying maximum angle tolerance  $\psi_{\text{tol}}$ 
1  $\phi \in \langle 0, \pi/2, \pi, 3\pi/2 \rangle$ ; /* initialize  $\phi$  in the four cardinal directions */
2  $\psi_{\text{max}} \leftarrow \pi$ ; count  $\leftarrow 0$ ; /* initialize */
3  $(\hat{\theta}_1, \hat{\theta}_2) \leftarrow \text{argmax}\{\log L(\theta_1, \theta_2)\}$ ; /* maximum log likelihood arguments are the MLEs */
4 while ( $\psi_{\text{max}} > \psi_{\text{tol}} \cap (\text{count} < \text{count}_{\text{max}})$ ) do
5     count  $\leftarrow$  count + 1; /* increment counter */
6      $m \leftarrow \text{length}(\phi)$ ; /* number of angles in  $\phi$  vector */
7     for  $i$  from 1 to  $m$  do
8          $(x_i, y_i) \leftarrow g(\phi_i, \alpha)$ ; /* CR point corresponding to  $\phi_i$  */
9      $p \leftarrow \langle (x_1, y_1), (x_2, y_2), \dots, (x_m, y_m) \rangle$ ; /* all current CR points */
10     $x_{\text{range}} \leftarrow \max\{x_1, x_2, \dots, x_m\} - \min\{x_1, x_2, \dots, x_m\}$ ; /* horizontal axis range */
11     $y_{\text{range}} \leftarrow \max\{y_1, y_2, \dots, y_m\} - \min\{y_1, y_2, \dots, y_m\}$ ; /* vertical axis range */
12     $s \leftarrow (x_{\text{range}})/(y_{\text{range}})$ ; /* (x range):(y range) ratio of CR plot */
13    for  $i$  from 1 to  $m$  do
14         $(x_i, y'_i) \leftarrow (x_i, s \cdot y_i)$ ; /* transformation enabling apparent angle calculations */
15     $p' \leftarrow \langle (x_1, y'_1), (x_2, y'_2), \dots, (x_m, y'_m) \rangle$ ; /* transformed CR points */
16     $p' \leftarrow \langle (x_m, y'_m), p', (x_1, y'_1) \rangle$ ; /* repeat end-points to ease analysis */
17    for  $i$  from 2 to  $(m + 1)$  do
18         $l_1 \leftarrow d(p'_{i-1}, p'_i)$ ; /* preceding segment distance */
19         $l_2 \leftarrow d(p'_i, p'_{i+1})$ ; /* next segment distance */
20         $l_3 \leftarrow d(p'_{i-1}, p'_{i+1})$ ; /* preceding-to-next-point segment distance */
21         $\psi_{i-1} \leftarrow \pi - \arccos(l_1^2 + l_2^2 - l_3^2)/(2 \cdot l_1 \cdot l_2)$ ; /* apparent angle (law of cosines) */
22     $\psi_{\text{max}} \leftarrow \max\{\psi_1, \psi_2, \dots, \psi_m\}$ ; /* current apparent maximum angle */
23     $\psi \leftarrow \langle \psi_m, \psi_1, \psi_2, \dots, \psi_m, \psi_1 \rangle$ ; /* repeat end-values to ease analysis */
24     $p \leftarrow \langle (x_m, y_m), p, (x_1, y_1) \rangle$ ; /* repeat end-points to ease analysis */
25     $\phi_{\text{new}} \leftarrow \{\}$ ; /* initialize */
26    for  $i$  from 2 to  $(m + 1)$  do
27        if  $\psi_i > \psi_{\text{tol}}$  then
28            if  $\psi_{i-1} > \psi_{\text{tol}}$  then
29                 $\phi_{\text{new}} \leftarrow \langle \phi_{\text{new}}, \phi \text{ from MLE to mid-point of } p_i \text{ and } p_{i-1} \rangle$ ; /* add  $\phi$  before */
30            if  $\psi_{i+1} > \psi_{\text{tol}}$  then
31                 $\phi_{\text{new}} \leftarrow \langle \phi_{\text{new}}, \phi \text{ from MLE to mid-point of } p_i \text{ and } p_{i+1} \rangle$ ; /* add  $\phi$  after */
32            if ( $\psi_{i-1} < \psi_{\text{tol}} \cap (\psi_{i+1} < \psi_{\text{tol}})$ ) then
33                 $\phi_{\text{new}} \leftarrow \langle \phi_{\text{new}}, \phi \text{ from MLE to mid-point of } p_i \text{ and } p_{i-1},$  /* add  $\phi$  before */
34                 $\phi \text{ from MLE to mid-point of } p_i \text{ and } p_{i+1} \rangle$ ; /* add  $\phi$  after */
35     $\phi \leftarrow \text{sort}(\langle \phi, \phi_{\text{new}} \rangle)$ ; /* augment  $\phi$  with  $\phi_{\text{new}}$  and sort its result in ascending order */

```

**Appendix C. Jump-Center Repair Heuristic**

The short description below summarizes Algorithm 3, which extends the smoothing search heuristic to repair otherwise radially inaccessible regions of its confidence region boundary. Its pseudo-code follows execution of Algorithm 2, so Algorithm 2 inputs and outputs are accessible to it.

Line 1 runs the smoothing search heuristic, so its inputs and outputs become available to Algorithm 3. The condition in Line 2 is true if Algorithm 2 terminates before satisfying its maximum degree tolerance  $\psi_{\text{tol}}$ , indicating inaccessible regions of the CR boundary. Line 3 identifies quadrants, with respect to the MLE, where inaccessible regions exist, and the loop from Lines 4–34 sequentially addresses repairs to those regions. Line 4 loops through the four quadrants relative to the MLE. Quadrants requiring repairs enter the if statement on Line 5.

Lines 6–13 identify three noteworthy reference values related to the CR points bordering its inaccessible region: the index value of its point nearer the MLE, and the vertical axis values of both points. Lines 14–25 use these values to identify an appropriate angle from the MLE to locate the jump-center. This angle depends on: the quadrant (relative to the MLE) of the inaccessible region, and the orientation of the inaccessible region (if it is above or below the line segment marking its border, reference Figure 9). The latter of these variables is known by comparing the vertical axis CR value on adjacent points of both sides of its inaccessible region. With these three variables, we identify whether the line segment from the MLE to the jump-center must pass through a horizontal segment to the left or right of the nearer CR inaccessible region border point, or a vertical segment above or below that point. That segment length (the feasible length within the CR that the jump-center angle passes through) is given by the variable gap

**Algorithm 3:** Jump-Center Repairs to Algorithm 2 for Inaccessible Confidence Region Boundary Points

```

input : Algorithm 2 inputs:  $\alpha$ ,  $\psi_{tol}$ ,  $\text{count}_{max}$ ,  $\hat{\theta}_1$ ,  $\hat{\theta}_2$ ,  $g(\phi_i, \alpha)$ ,  $d((x_i, y_i), (x_j, y_j))$ 
Algorithm 2 output:  $\phi_{new}$ ,  $p_i = (x_i, y_i)$  for  $i = 1, 2, \dots, m$  coordinate pairs
 $\alpha_{jump}$  : jump-center significance level, "uphill" of  $\alpha < \alpha_{jump}$ 
 $b$  : bi-section percentage; determines the angle from the MLE where the jump-center will locate
output:  $p_i = (x_i, y_i)$  for  $i = 1, 2, \dots, m$  coordinate pairs enclosing a  $100(1 - \alpha)\%$  confidence region (CR) for  $\theta_1$  and  $\theta_2$  satisfying maximum angle tolerance
 $\psi_{tol}$ , including jump-center repairs
1 run Algorithm 2;
2 if  $\text{count} = \text{count}_{max}$  then
3    $(\phi_I, \phi_{II}, \phi_{III}, \phi_{IV}) \leftarrow \phi_{new}$ ;
4   for  $q$  in  $\{I, II, III, IV\}$  do
5     if  $\phi_q \neq \{\}$  then
6       for  $k$  in 1 to  $\text{length}(\phi_q)$  do
7         if  $d((\hat{\theta}_1, \hat{\theta}_2), g(\phi_q[k], \alpha)) = \min\{d((\hat{\theta}_1, \hat{\theta}_2), g(\phi_q, \alpha))\}$  then
8            $\phi_{near} \leftarrow \phi_q[k]$ ;
9         for  $k$  in 1 to  $\text{length}(p)$  do
10          if  $\phi_{near} = \phi_k$  then
11             $i_{near} \leftarrow k$ ;
12           $y_{i_{near}-1} \leftarrow p_{i_{near}-1}[2]$ ;
13           $y_{i_{near}+1} \leftarrow p_{i_{near}+1}[2]$ ;
14          if  $((q=I) \cap (y_{i_{near}-1} < y_{i_{near}+1})) \cup ((q=II) \cap (y_{i_{near}-1} > y_{i_{near}+1}))$  then
15             $\text{gap} \leftarrow y\text{-range in CR above } y_{i_{near}}$ ;
16             $\phi_{jump} \leftarrow \text{angle from MLE to } (x_{i_{near}}, y_{i_{near}} + b \cdot \text{gap})$ ;
17          if  $((q=III) \cap (y_{i_{near}-1} > y_{i_{near}+1})) \cup ((q=IV) \cap (y_{i_{near}-1} < y_{i_{near}+1}))$  then
18             $\text{gap} \leftarrow y\text{-range in CR below } y_{i_{near}}$ ;
19             $\phi_{jump} \leftarrow \text{angle from MLE to } (x_{i_{near}}, y_{i_{near}} - b \cdot \text{gap})$ ;
20          if  $(q \in \{I, IV\}) \cap (y_{i_{near}-1} > y_{i_{near}+1})$  then
21             $\text{gap} \leftarrow x\text{-range in CR right of } x_{i_{near}}$ ;
22             $\phi_{jump} \leftarrow \text{angle from MLE to } (x_{i_{near}} + b \cdot \text{gap}, y_{i_{near}})$ ;
23          if  $(q \in \{II, III\}) \cap (y_{i_{near}-1} < y_{i_{near}+1})$  then
24             $\text{gap} \leftarrow x\text{-range in CR left of } x_{i_{near}}$ ;
25             $\phi_{jump} \leftarrow \text{angle from MLE to } (x_{i_{near}} - b \cdot \text{gap}, y_{i_{near}})$ ;
26           $p_{jump} \leftarrow g(\phi_{jump}, \alpha_{jump})$ ;
27           $j \leftarrow \text{run Algorithm 2 with } p_{jump} \text{ replacing the MLE}$ ;
28           $j_{repair} \subset j$ ;
29          if  $((q \in \{I, II\}) \cap (y_{i_{near}-1} < y_{i_{near}+1})) \cup ((q \in \{III, IV\}) \cap (y_{i_{near}-1} > y_{i_{near}+1}))$  then
30             $p_{add} \leftarrow p_{i_{near}}$ ;
31          else
32             $p_{add} \leftarrow p_{i_{near}-1}$ ;
33           $p \leftarrow (p_1, \dots, p_{add}, j_{repair}, p_{add+1}, \dots, m)$ ;
34           $m \leftarrow \text{length}(p)$ ;

```

in Lines 15, 18, 21, and 24. The specific jump-center angle is then given in Lines 16, 19, 22, and 25 using the input variable  $b$  to discern where along the feasible  $\text{gap}$  segment the jump-center angle will pass. Line 26 identifies the jump-center coordinates. They are then used in Line 27 to recursively call Algorithm 2 with one exception to its original parameterization:  $p_{jump}$  replaces the MLE as the point of reference where radial azimuths are taken. The subset of points within its results

that fill in the previously inaccessible region are isolated in Line 28. Lines 29–32 identify the location within the original  $p$  CR boundary points to insert the jump-center repair points following, which depends on the inaccessible region orientation (reference Figure 9). Line 33 then integrates those results into the previous CR solution in the proper sequence. Finally, Line 34 updates the length of the CR points solution.