Fitting algorithms

# Circle fitting[[1]](#footnote-1)

## Geometrical fits

Minimizing the geometric distance from the circle to the data points.

These algorithms are regarded as the most accurate circle fits, but computationally demanding. They require iterative schemes that increase computational load, and are prone to occasional divergence.

The two main members of this family are:

1. **Orthogonal Distance Regression (ODR):** minimizes the function:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | | (1) |
|  |  |  |

The algorithm depends heavily on initial guess, and requires up to hundreds of iterations to converge. There is a risk of converging to a local minimum or diverging.

1. **Pratt scheme:** describe the circle as:

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

With the constraints that and , so that:

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

Pratt suggested taking .

## Algebraic fits

Minimize various approximate (or ‘algebraic’) distances. They are non-iterative and therefore fast and less computationally heavy, but also less accurate.

1. **Kasa fit:** minimizes the distance from point to the a circle with radius *R*. The minimized function is:

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

Where is the geometric distance.

This is the fastest fit, but inaccurate for incomplete arcs and large circles.

1. **Pratt fit:** provides a better approximation to the geometrical distance, minimizing the function:

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

Again, under the constraint .

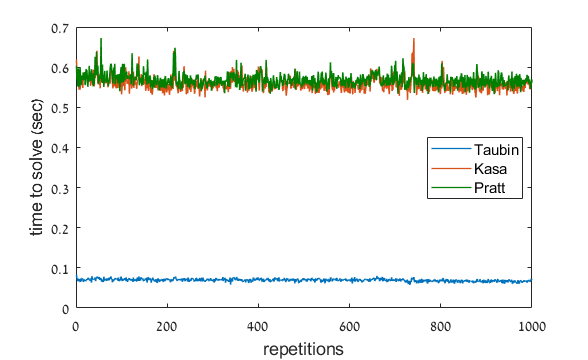
1. **Taubin fit:** Taubin suggested fitting equation (5) under a new constraint:

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

Where .

The Pratt and Taubin fits are more stable and accurate than Kasa fit.

The Taubin fit is statistically more accurate than that of Pratt.



Therefore the Taubin fit is the most appropriate.

# Ellipse fitting

## Describing an ellipse

An ellipse has five parameters.

**Conic**: An ellipse is the set of points that satisfy:

subject to the constraints

**Parametric**: can be written as





1. Mainly derived from:

   Al-Sharadqah, A. & Chernov, N. Error analysis for circle fitting algorithms. *Electron. J. Stat.* **3,** 886–911 (2009). [↑](#footnote-ref-1)