TK Solver Case Study – Finding the Best-Fitting Circle

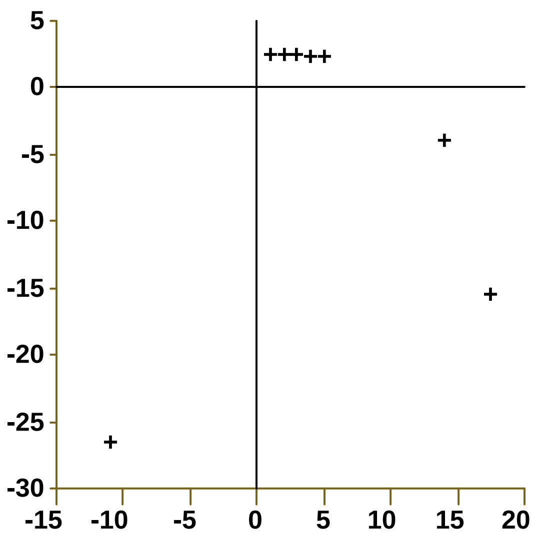
Problem:
Given a set of coordinates, determine the center and radius of the circle that fits the points most closely.

Assumptions:
The x and y coordinates are given as two lists of values.
The goodness of fit is defined as the sum of squared deviations between the radius of the circle and the distance to the coordinates from the center. This should be minimized.

Sample Data:
Here is a table of coordinates for testing purposes.

Element	x	y
1	1	2.4
2	2	2.45
3	3	2.5
4	4	2.35
5	5	2.3
6	17.5	-15.5
7	14	-4
8	-11	-26.5

Here is a plot of those coordinates.



Solution:
Create a function that defines the fit of the circle with the points. A procedure function can be used. The function has 3 inputs – h, k, and r – representing the center coordinates and the radius of the circle and one output, s, representing the sum of squared deviations.

Statement
s = 0
for i = 1 to length('x)
s = s + (SQRT(('x[i]-h)^2+('y[i]-k)^2)-r)^2
next i

The procedure loops through the coordinates, computing the distances from the center, and summing the squared deviations.

A single equation on the rule sheet can be used to reference this procedure.

Rule
s = f(h,k,r)

Test the function with sample inputs for h, k, and r. Based on the plot of the coordinates, we might try 2 and -12 for the center coordinates and 15 for the radius. Solving (F9) produces the following result.

St	Input	Name	Output	Unit	Comment
	2	h			x-coordinate of center
	-12	k			y-coordinate of center
	15	r			radius
		s	22.3974724		sum of squares

The goal is to determine the values of h, k, and r that result in a minimum value of s. If you have the Premium Version of TK Solver, you can use the Optimizer with s as the Objective Variable and h, k, and r as the Decision Variables. Here is the resulting solution.

St	Input	Name	Output	Unit	Comment
	.673061622	h			x-coordinate of center
	-14.276402	k			y-coordinate of center
	16.8857038	r			radius
		s	.125718916		sum of squares

If you do not have the Optimizer, you can enter three more equations, setting the partial derivatives of h, k, and r equal to zero and let the TK Iterative Solver find the solution. Here are those additional equations.

Rule
s = f(h+.0001,k,r)
s = f(h,k+.0001,r)
s = f(h,k,r+.0001)

The points at which all four of these equations are true are minimum and maximum values of s. The choices for the initial guesses of h, k and r are important in arriving at the desired solution. Fortunately, we have the plot for rough estimates. We might use guesses of 2 and -12 for the center coordinates and 15 for the radius.

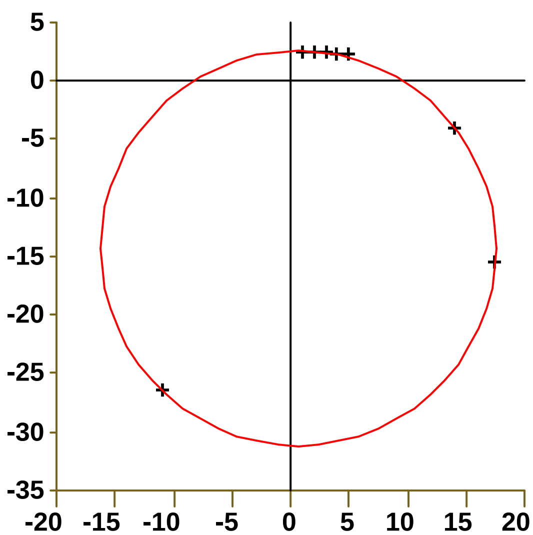
Here is TK's solution. It is very similar to that found by the Optimizer.

St	Input	Name	Output	Unit	Comment
		h	.673054027		x-coordinate of center
		k	-14.276418		y-coordinate of center
		r	16.8856649		radius
		s	.125718937		sum of squares

Finally, we can append a plot of the circle to the sample points using another procedure function. Here are the required statements, assuming that h, k, and r are passed into the function as Parameter Variables.

Statement
n = length('x)   ; list of x-coordinates
call blankm('xp,'yp)   ; clear previous plot data
call listcopy('x,'xp)   ; copy coordinates to plot
t = 0
for i = 1 to 61
('xp[i+n],'yp[i+n]) = ptord(r,t) + (h,k)   ; generate circle points
t = t + 6
next i

Adding call plot() to the rule sheet and using xp, y, and yp as the plot lists results in the following plot.



The fit appears to be ok.