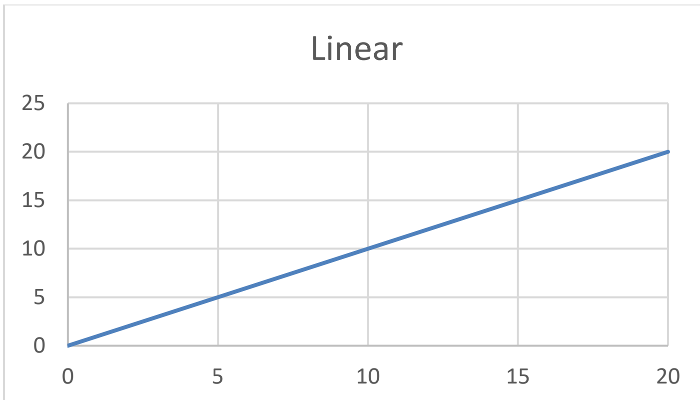
Microsoft Excel

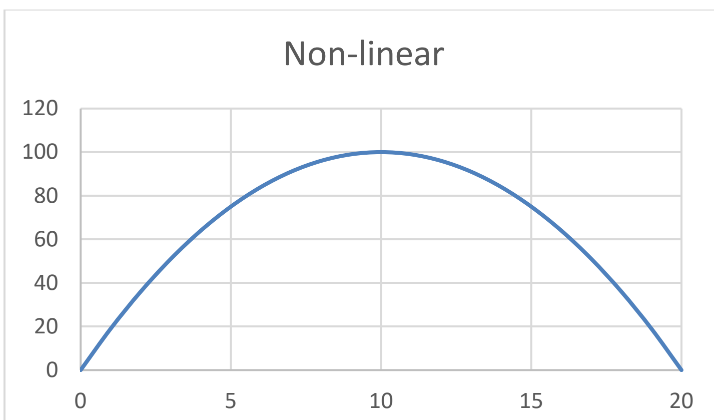
Solver Optimization Supplementary Material

Solving Non-linear Problems in Solver

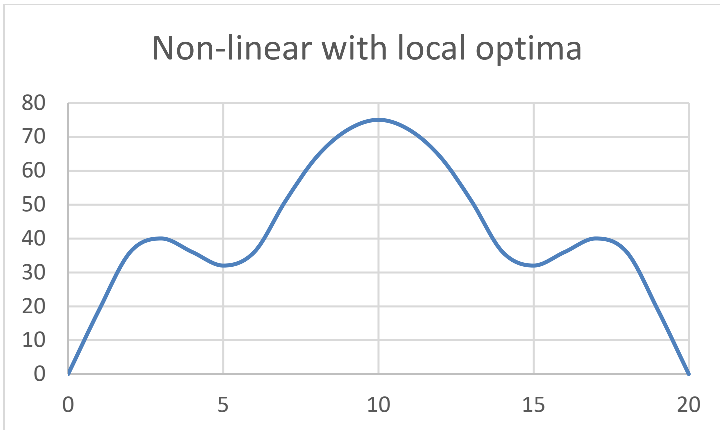
The simplest Objective curves will be linear. The Simplex method is sufficient when the function is linear.



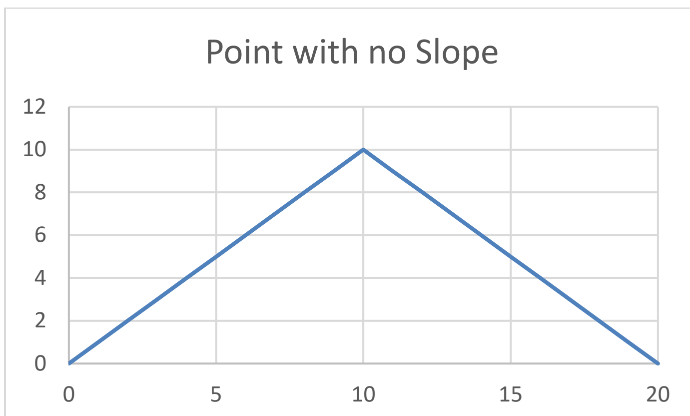
Non-linear equations require a non-linear solution. GRG Nonlinear can solve this.



Non-linear with local optima require multiple starting points. GRG Nonlinear with multi-start works here.



Equation with point having no slope. Evolutionary must be used here.



Summary of technique selection

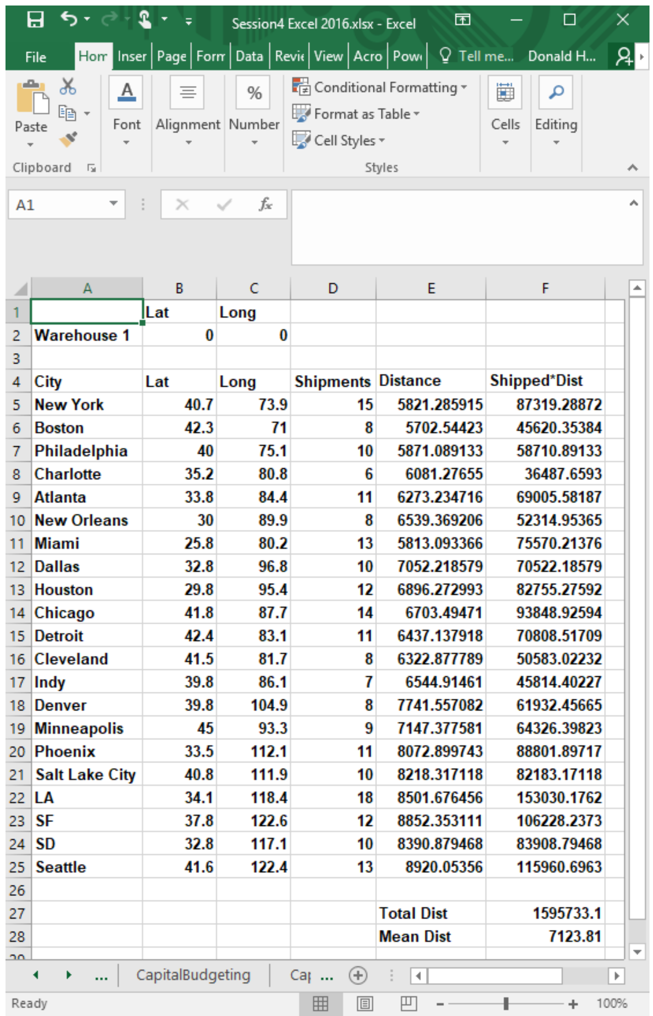
The following table summarizes which techniques can solve each type of problem.

	Solution Technique			
Problem	Simplex	GRG Non-linear	GRG Non-linear with multi-start	Evolutionary
Linear	Yes	Yes	Yes	Yes
Nonlinear, one optimum	No	Yes	Yes	Yes
Nonlinear, multiple optima	No	No	Yes	Yes
Curve has point with no slope	No	No	No	Yes
Speed	Fastest	Fast	Slow	Slowest

Warehouse Location – One Warehouse

You can use solver to find the optimal location for a warehouse, which minimizes the shipping distances for all shipments. We will try to find the optimal location for one warehouse to minimize shipment miles across the country, then expand to two warehouses.

1. For the first example in locating one warehouse, use the WarehouseLocation1 spreadsheet.



	A	B	C	D	E	F
1		Lat	Long			
2	Warehouse 1	0	0			
3						
4	City	Lat	Long	Shipments	Distance	Shipped*Dist
5	New York	40.7	73.9	15	5821.285915	87319.28872
6	Boston	42.3	71	8	5702.54423	45620.35384
7	Philadelphia	40	75.1	10	5871.089133	58710.89133
8	Charlotte	35.2	80.8	6	6081.27655	36487.6593
9	Atlanta	33.8	84.4	11	6273.234716	69005.58187
10	New Orleans	30	89.9	8	6539.369206	52314.95365
11	Miami	25.8	80.2	13	5813.093366	75570.21376
12	Dallas	32.8	96.8	10	7052.218579	70522.18579
13	Houston	29.8	95.4	12	6896.272993	82755.27592
14	Chicago	41.8	87.7	14	6703.49471	93848.92594
15	Detroit	42.4	83.1	11	6437.137918	70808.51709
16	Cleveland	41.5	81.7	8	6322.877789	50583.02232
17	Indy	39.8	86.1	7	6544.91461	45814.40227
18	Denver	39.8	104.9	8	7741.557082	61932.45665
19	Minneapolis	45	93.3	9	7147.377581	64326.39823
20	Phoenix	33.5	112.1	11	8072.899743	88801.89717
21	Salt Lake City	40.8	111.9	10	8218.317118	82183.17118
22	LA	34.1	118.4	18	8501.676456	153030.1762
23	SF	37.8	122.6	12	8852.353111	106228.2373
24	SD	32.8	117.1	10	8390.879468	83908.79468
25	Seattle	41.6	122.4	13	8920.05356	115960.6963
26						
27					Total Dist	1595733.1
28					Mean Dist	7123.81

2. Each city is identified with latitude and longitude, the number of shipments going to that city, calculated distance from the city to the warehouse, and shipping miles (Shipped*Dist) for each city
3. The goal is to minimize the Total Distance by finding the best location for warehouse 1.

4. To calculate the distance from each city to the warehouse, we will use the Pythagorean theorem:

   $C^2 = A^2 + B^2$

5. Or, taking the square root of both sides:

   $C = SQRT(A^2 + B^{2)}$

6. For example, to calculate the distance from Dallas to Columbus:

7. We can approximate the distance A:

   A = latitude of Columbus – latitude of Dallas

8. We can approximate the distance B:

   B = longitude of Columbus – longitude of Dallas

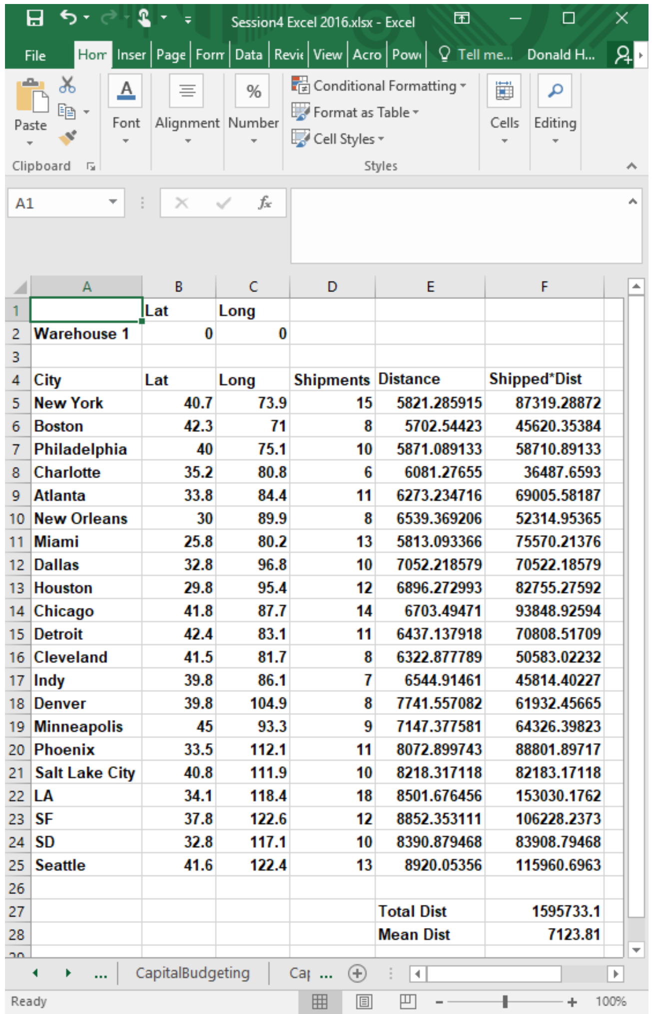
9. Then the distance C is:

   $C = SQRT((lat(Columbus)-lat(Dallas))^2 + (long(Columbus)-long(Dallas))^2)$

10. One degree of latitude or longitude is approximately 69 miles, so the distance in miles is:

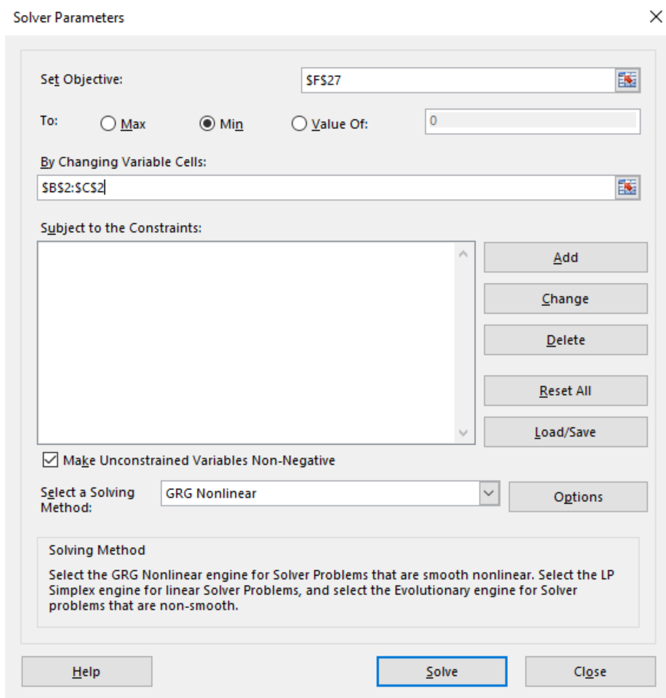
    $Distance = 69*SQRT((lat(Columbus)-lat(Dallas))^2 + (long(Columbus)-long(Dallas))^2)$

11. In column E, we calculate the distance from the warehouse to each city.
12. In column F, calculate the number of shipments * distance, so we have total miles driven.
13. In F27, create total distance for all cities

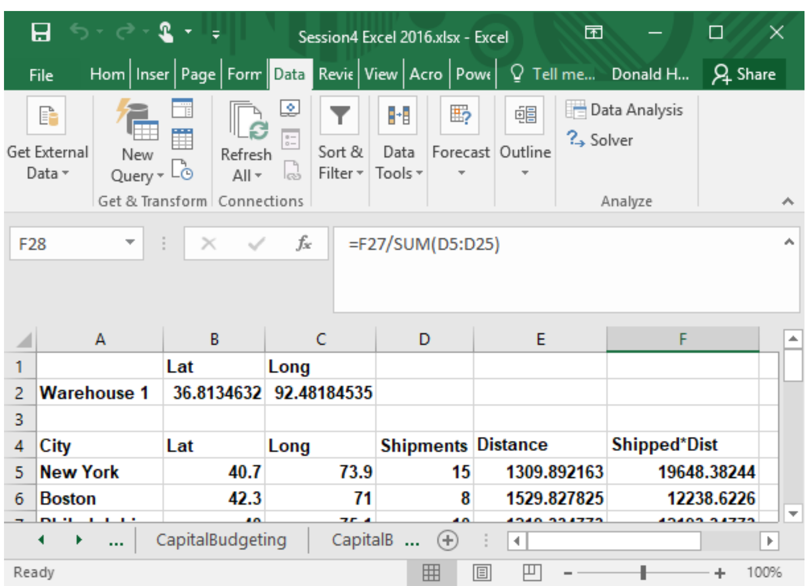


	A	B	C	D	E	F
1		Lat	Long			
2	Warehouse 1	0	0			
3						
4	City	Lat	Long	Shipments	Distance	Shipped*Dist
5	New York	40.7	73.9	15	5821.285915	87319.28872
6	Boston	42.3	71	8	5702.54423	45620.35384
7	Philadelphia	40	75.1	10	5871.089133	58710.89133
8	Charlotte	35.2	80.8	6	6081.27655	36487.6593
9	Atlanta	33.8	84.4	11	6273.234716	69005.58187
10	New Orleans	30	89.9	8	6539.369206	52314.95365
11	Miami	25.8	80.2	13	5813.093366	75570.21376
12	Dallas	32.8	96.8	10	7052.218579	70522.18579
13	Houston	29.8	95.4	12	6896.272993	82755.27592
14	Chicago	41.8	87.7	14	6703.49471	93848.92594
15	Detroit	42.4	83.1	11	6437.137918	70808.51709
16	Cleveland	41.5	81.7	8	6322.877789	50583.02232
17	Indy	39.8	86.1	7	6544.91461	45814.40227
18	Denver	39.8	104.9	8	7741.557082	61932.45665
19	Minneapolis	45	93.3	9	7147.377581	64326.39823
20	Phoenix	33.5	112.1	11	8072.899743	88801.89717
21	Salt Lake City	40.8	111.9	10	8218.317118	82183.17118
22	LA	34.1	118.4	18	8501.676456	153030.1762
23	SF	37.8	122.6	12	8852.353111	106228.2373
24	SD	32.8	117.1	10	8390.879468	83908.79468
25	Seattle	41.6	122.4	13	8920.05356	115960.6963
26						
27					Total Dist	1595733.1
28					Mean Dist	7123.81

14. To run solver, click on the Data tab, then Solver
15. The objective is to minimize total distance, so enter F27 in Set Objective
16. We want to change the warehouse location, so set By Change Variable Cells to B2:C2, the latitude and longitude for the warehouse
17. Select a solving method of GRG nonlinear.
18. Click Solve

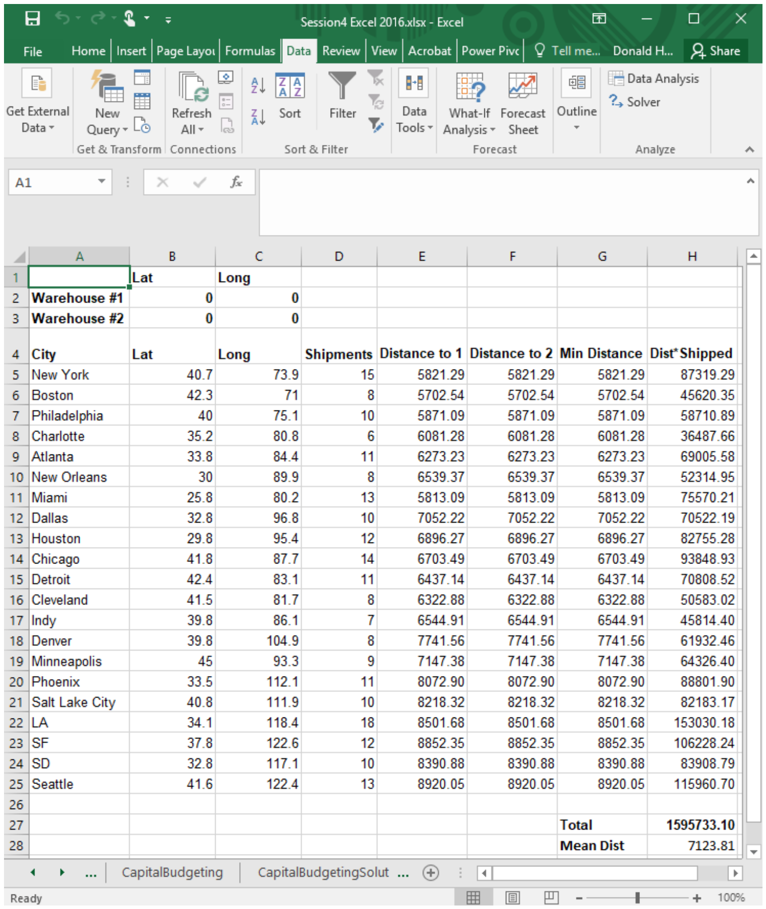


19. The solution is 36.81 N, 92.48 W. Use Google Maps to identify the location



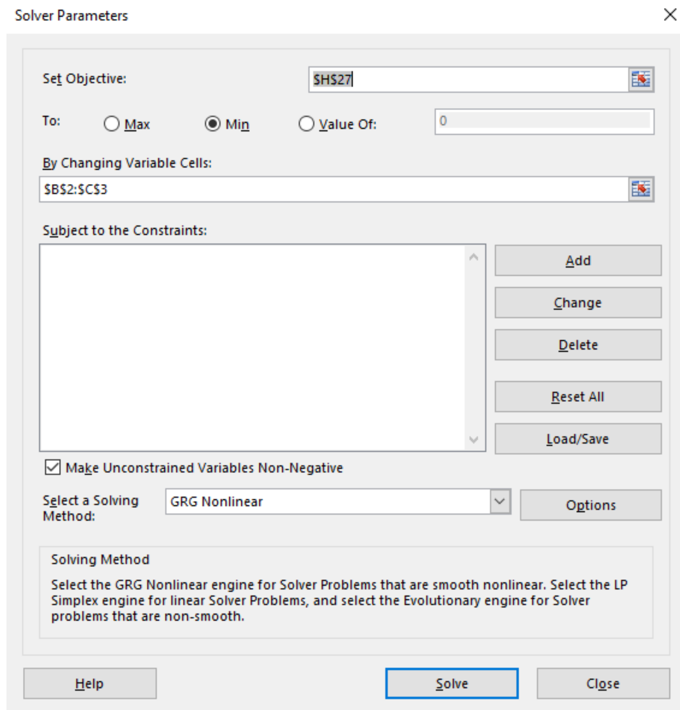
Warehouse Location – Two Warehouses

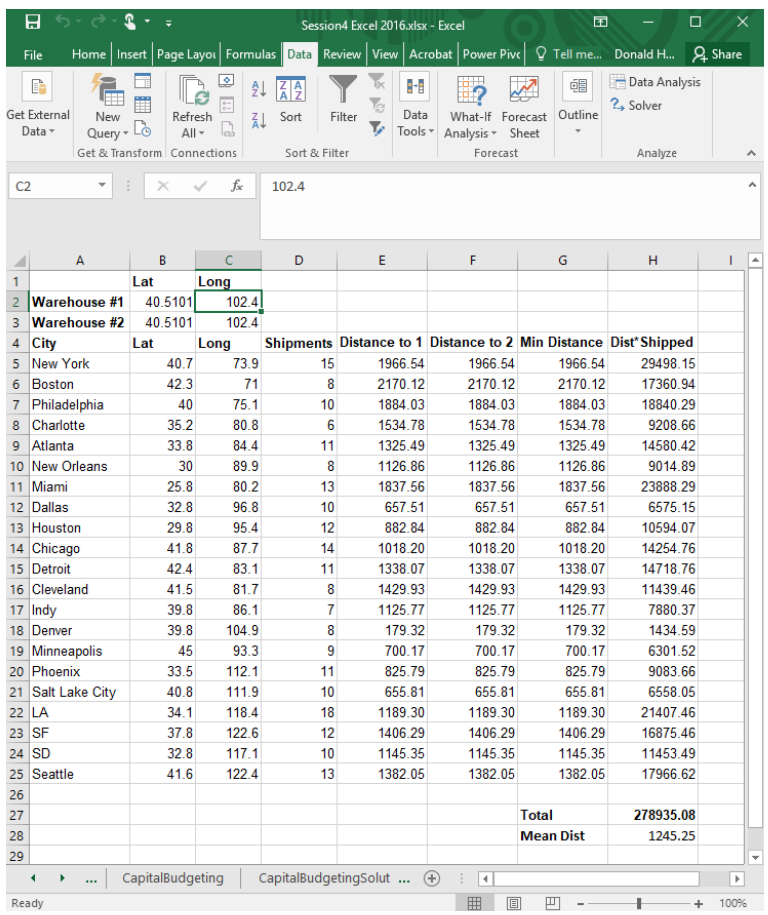
Now, assume that we will locate two warehouses.



	A	B	C	D	E	F	G	H
1		**Lat**	**Long**					
2	**Warehouse #1**	**0**	**0**					
3	**Warehouse #2**	**0**	**0**					
4	**City**	**Lat**	**Long**	**Shipments**	**Distance to 1**	**Distance to 2**	**Min Distance**	**Dist*Shipped**
5	New York	40.7	73.9	15	5821.29	5821.29	5821.29	87319.29
6	Boston	42.3	71	8	5702.54	5702.54	5702.54	45620.35
7	Philadelphia	40	75.1	10	5871.09	5871.09	5871.09	58710.89
8	Charlotte	35.2	80.8	6	6081.28	6081.28	6081.28	36487.66
9	Atlanta	33.8	84.4	11	6273.23	6273.23	6273.23	69005.58
10	New Orleans	30	89.9	8	6539.37	6539.37	6539.37	52314.95
11	Miami	25.8	80.2	13	5813.09	5813.09	5813.09	75570.21
12	Dallas	32.8	96.8	10	7052.22	7052.22	7052.22	70522.19
13	Houston	29.8	95.4	12	6896.27	6896.27	6896.27	82755.28
14	Chicago	41.8	87.7	14	6703.49	6703.49	6703.49	93848.93
15	Detroit	42.4	83.1	11	6437.14	6437.14	6437.14	70808.52
16	Cleveland	41.5	81.7	8	6322.88	6322.88	6322.88	50583.02
17	Indy	39.8	86.1	7	6544.91	6544.91	6544.91	45814.40
18	Denver	39.8	104.9	8	7741.56	7741.56	7741.56	61932.46
19	Minneapolis	45	93.3	9	7147.38	7147.38	7147.38	64326.40
20	Phoenix	33.5	112.1	11	8072.90	8072.90	8072.90	88801.90
21	Salt Lake City	40.8	111.9	10	8218.32	8218.32	8218.32	82183.17
22	LA	34.1	118.4	18	8501.68	8501.68	8501.68	153030.18
23	SF	37.8	122.6	12	8852.35	8852.35	8852.35	106228.24
24	SD	32.8	117.1	10	8390.88	8390.88	8390.88	83908.79
25	Seattle	41.6	122.4	13	8920.05	8920.05	8920.05	115960.70
26								
27							**Total**	**1595733.10**
28							**Mean Dist**	7123.81

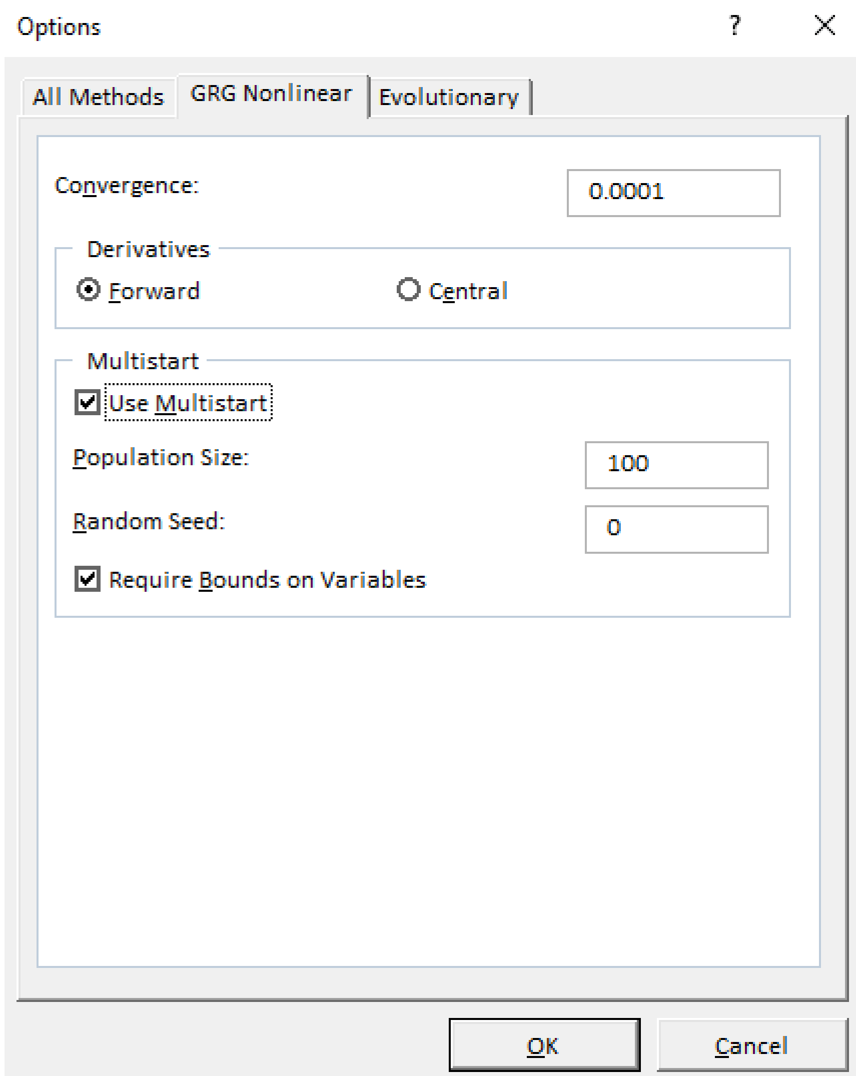
1. The latitude and longitude for our two warehouses are at the top of the screen.
2. The only additions that we have are two distance calculations (city to warehouse 1 and city to warehouse 2) and the minimum distance of the two. Our goal is to minimize total distance. In this case, we will assume that each city will receive shipments from the warehouse that is closest
3. To run solver, click on the Data tab, then Solver
4. The objective is to minimize total distance, so enter F27 in Set Objective
5. We want to change the warehouse location, so set By Change Variable Cells to B2:C3, the latitude and longitude for the warehouses
6. Select a solving method of GRG nonlinear.
7. Click Solve
8. Does the result make sense? The two warehouses are located in the same latitude and longitude.
9. Solver is stuck in a local solution



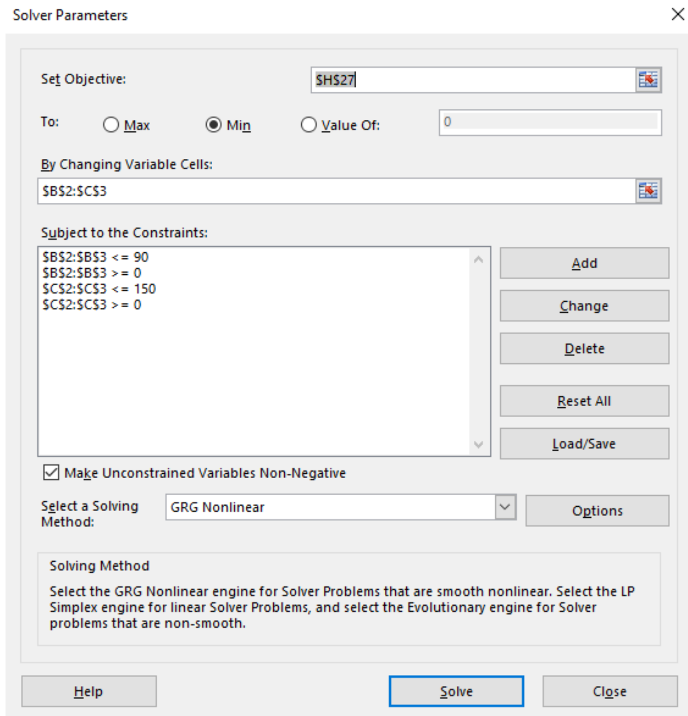


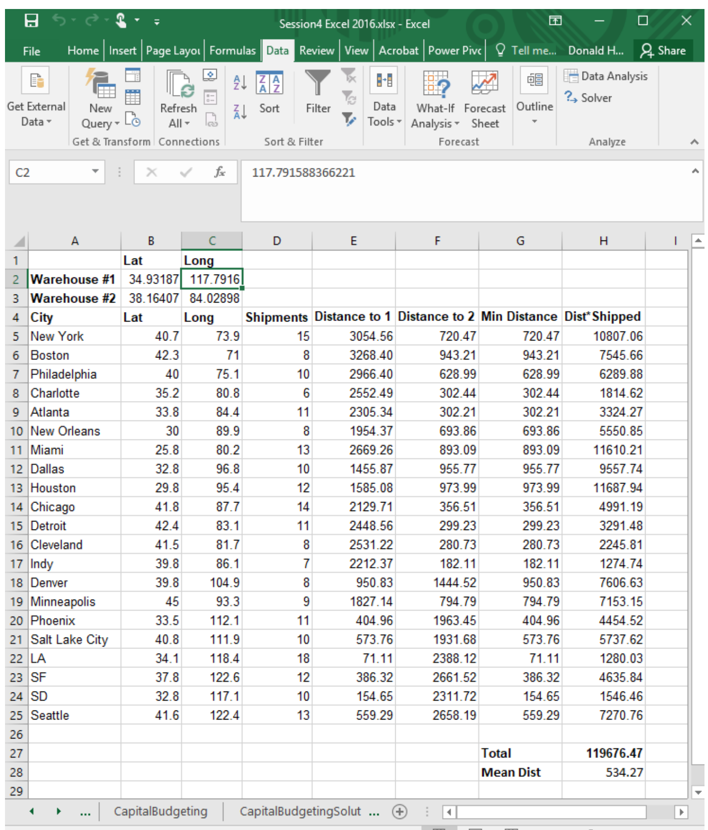
	A	B	C	D	E	F	G	H
1		Lat	Long					
2	**Warehouse #1**	40.5101	102.4					
3	**Warehouse #2**	40.5101	102.4					
4	**City**	**Lat**	**Long**	**Shipments**	**Distance to 1**	**Distance to 2**	**Min Distance**	**Dist*Shipped**
5	New York	40.7	73.9	15	1966.54	1966.54	1966.54	29498.15
6	Boston	42.3	71	8	2170.12	2170.12	2170.12	17360.94
7	Philadelphia	40	75.1	10	1884.03	1884.03	1884.03	18840.29
8	Charlotte	35.2	80.8	6	1534.78	1534.78	1534.78	9208.66
9	Atlanta	33.8	84.4	11	1325.49	1325.49	1325.49	14580.42
10	New Orleans	30	89.9	8	1126.86	1126.86	1126.86	9014.89
11	Miami	25.8	80.2	13	1837.56	1837.56	1837.56	23888.29
12	Dallas	32.8	96.8	10	657.51	657.51	657.51	6575.15
13	Houston	29.8	95.4	12	882.84	882.84	882.84	10594.07
14	Chicago	41.8	87.7	14	1018.20	1018.20	1018.20	14254.76
15	Detroit	42.4	83.1	11	1338.07	1338.07	1338.07	14718.76
16	Cleveland	41.5	81.7	8	1429.93	1429.93	1429.93	11439.46
17	Indy	39.8	86.1	7	1125.77	1125.77	1125.77	7880.37
18	Denver	39.8	104.9	8	179.32	179.32	179.32	1434.59
19	Minneapolis	45	93.3	9	700.17	700.17	700.17	6301.52
20	Phoenix	33.5	112.1	11	825.79	825.79	825.79	9083.66
21	Salt Lake City	40.8	111.9	10	655.81	655.81	655.81	6558.05
22	LA	34.1	118.4	18	1189.30	1189.30	1189.30	21407.46
23	SF	37.8	122.6	12	1406.29	1406.29	1406.29	16875.46
24	SD	32.8	117.1	10	1145.35	1145.35	1145.35	11453.49
25	Seattle	41.6	122.4	13	1382.05	1382.05	1382.05	17966.62
26								
27							**Total**	**278935.08**
28							**Mean Dist**	1245.25
29								

10. Now, let's use GRG Nonlinear with multiple start points
11. Click on solver
12. Next to GRG Nonlinear, click on Options
13. Click on the tab GRG Nonlinear
14. Check the box Use Multistart, then click OK
15. Note that it requires bounds on variables



16. Next add constraints that put bounds on the variables
17. In the Solver Parameters screen, under constraints, click Add
18. Add a constraint for B2:B3 >= 0
19. Add a constraint for B2:B3 <= 90
20. Add a constraint for C2:C3 >= 0
21. Add a constraint for C2:C3 <= 150
22. Click Solve
23. Use Google Maps to find the locations of the two warehouses



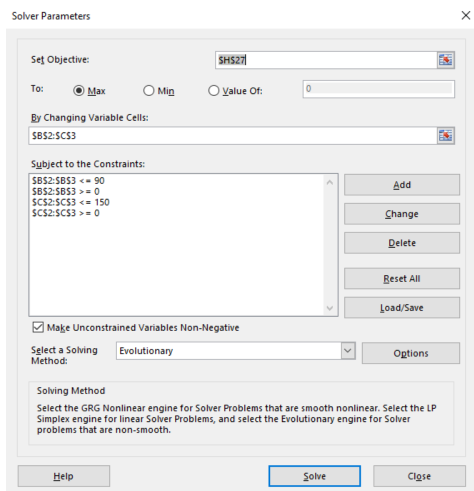


	A	B	C	D	E	F	G	H
1		Lat	Long					
2	Warehouse #1	34.93187	117.7916					
3	Warehouse #2	38.16407	84.02898					
4	City	Lat	Long	Shipments	Distance to 1	Distance to 2	Min Distance	Dist*Shipped
5	New York	40.7	73.9	15	3054.56	720.47	720.47	10807.06
6	Boston	42.3	71	8	3268.40	943.21	943.21	7545.66
7	Philadelphia	40	75.1	10	2966.40	628.99	628.99	6289.88
8	Charlotte	35.2	80.8	6	2552.49	302.44	302.44	1814.62
9	Atlanta	33.8	84.4	11	2305.34	302.21	302.21	3324.27
10	New Orleans	30	89.9	8	1954.37	693.86	693.86	5550.85
11	Miami	25.8	80.2	13	2669.26	893.09	893.09	11610.21
12	Dallas	32.8	96.8	10	1455.87	955.77	955.77	9557.74
13	Houston	29.8	95.4	12	1585.08	973.99	973.99	11687.94
14	Chicago	41.8	87.7	14	2129.71	356.51	356.51	4991.19
15	Detroit	42.4	83.1	11	2448.56	299.23	299.23	3291.48
16	Cleveland	41.5	81.7	8	2531.22	280.73	280.73	2245.81
17	Indy	39.8	86.1	7	2212.37	182.11	182.11	1274.74
18	Denver	39.8	104.9	8	950.83	1444.52	950.83	7606.63
19	Minneapolis	45	93.3	9	1827.14	794.79	794.79	7153.15
20	Phoenix	33.5	112.1	11	404.96	1963.45	404.96	4454.52
21	Salt Lake City	40.8	111.9	10	573.76	1931.68	573.76	5737.62
22	LA	34.1	118.4	18	71.11	2388.12	71.11	1280.03
23	SF	37.8	122.6	12	386.32	2661.52	386.32	4635.84
24	SD	32.8	117.1	10	154.65	2311.72	154.65	1546.46
25	Seattle	41.6	122.4	13	559.29	2658.19	559.29	7270.76
26								
27							Total	119676.47
28							Mean Dist	534.27
29								

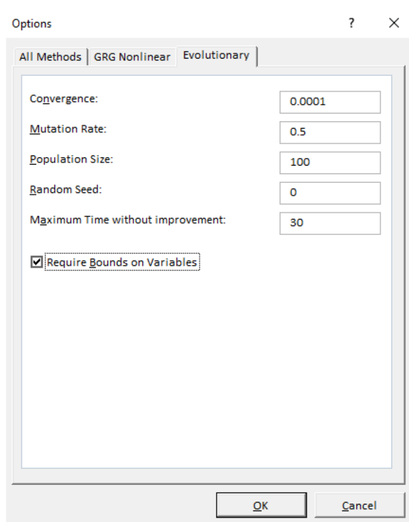
Evolutionary Solver

Run the same problem with Evolutionary

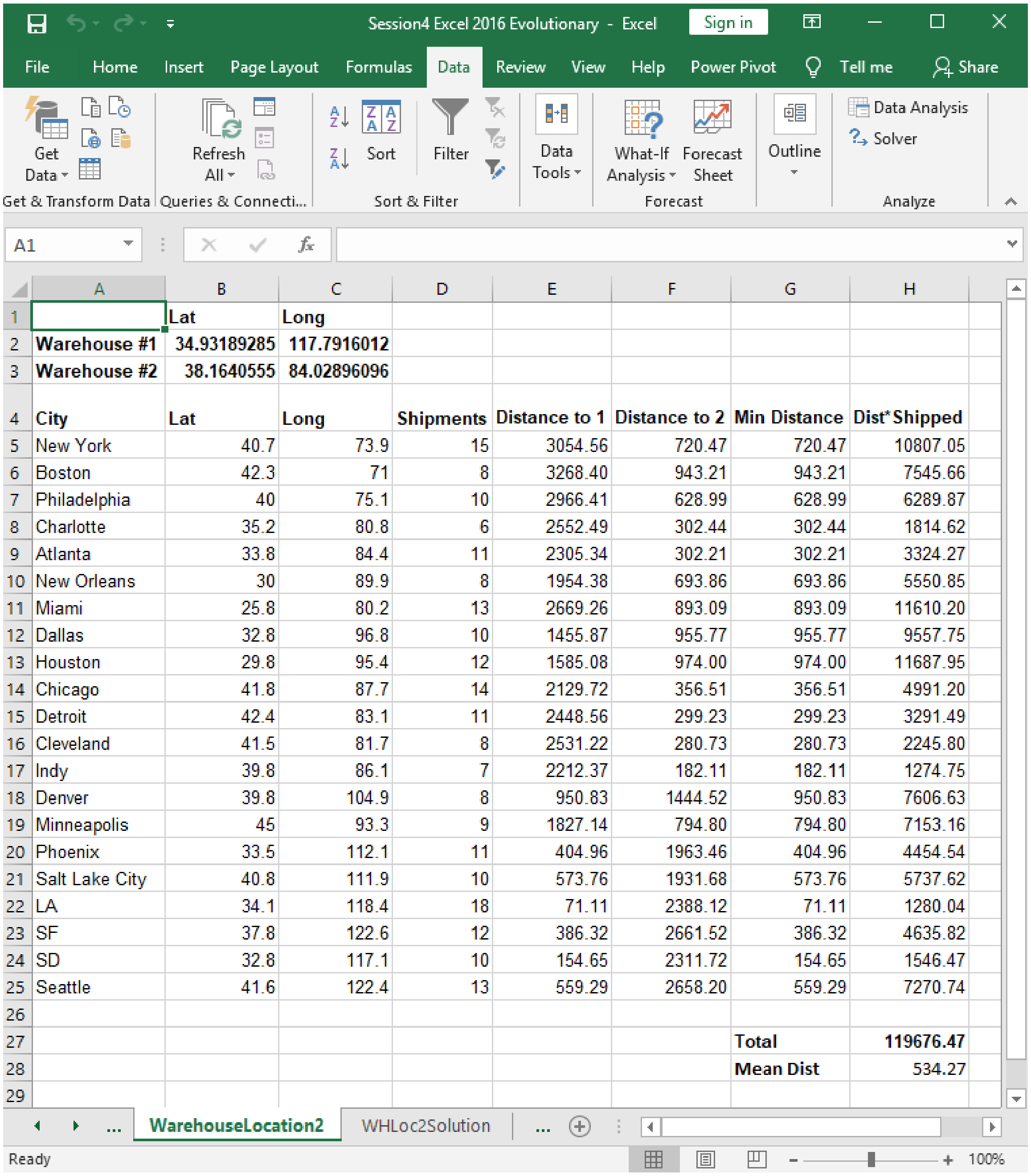
1. Set the objective to Total Distance (H27)
2. Set to Min so we minimize Total Distance
3. Set By Changing Variable Cells to B2:C3
4. Add a constraint for B2:B3 >= 0
5. Add a constraint for B2:B3 <= 90
6. Add a constraint for C2:C3 >= 0
7. Add a constraint for C2:C3 <= 150



8. Change Select a Solving Method to Evolutionary
9. Click on Options next to Evolutionary
10. Change Mutation Rate to 0.5
11. Check the box Require Bounds on Variables
12. Click OK



13. Solver will return the Evolutionary solution

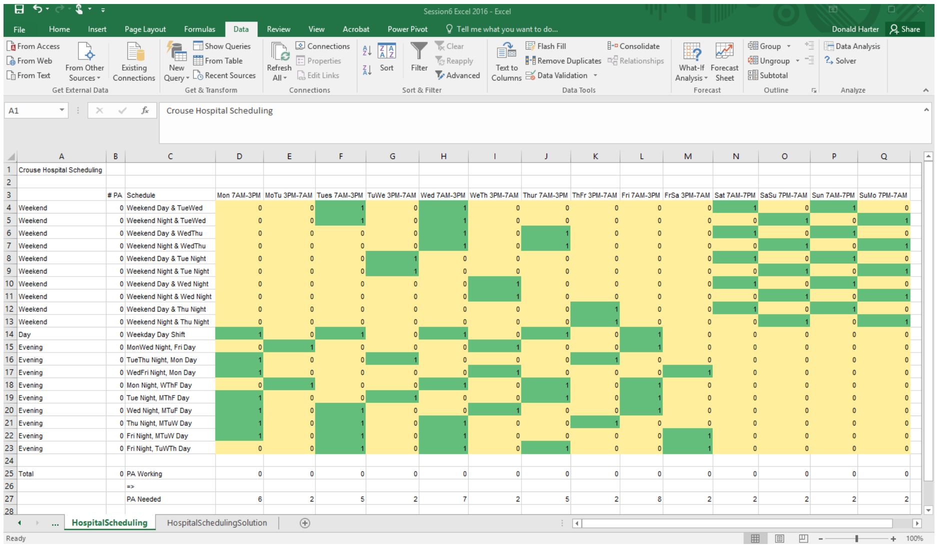


	A	B	C	D	E	F	G	H
1		Lat	Long					
2	Warehouse #1	34.93189285	117.7916012					
3	Warehouse #2	38.1640555	84.02896096					
4	City	Lat	Long	Shipments	Distance to 1	Distance to 2	Min Distance	Dist*Shipped
5	New York	40.7	73.9	15	3054.56	720.47	720.47	10807.05
6	Boston	42.3	71	8	3268.40	943.21	943.21	7545.66
7	Philadelphia	40	75.1	10	2966.41	628.99	628.99	6289.87
8	Charlotte	35.2	80.8	6	2552.49	302.44	302.44	1814.62
9	Atlanta	33.8	84.4	11	2305.34	302.21	302.21	3324.27
10	New Orleans	30	89.9	8	1954.38	693.86	693.86	5550.85
11	Miami	25.8	80.2	13	2669.26	893.09	893.09	11610.20
12	Dallas	32.8	96.8	10	1455.87	955.77	955.77	9557.75
13	Houston	29.8	95.4	12	1585.08	974.00	974.00	11687.95
14	Chicago	41.8	87.7	14	2129.72	356.51	356.51	4991.20
15	Detroit	42.4	83.1	11	2448.56	299.23	299.23	3291.49
16	Cleveland	41.5	81.7	8	2531.22	280.73	280.73	2245.80
17	Indy	39.8	86.1	7	2212.37	182.11	182.11	1274.75
18	Denver	39.8	104.9	8	950.83	1444.52	950.83	7606.63
19	Minneapolis	45	93.3	9	1827.14	794.80	794.80	7153.16
20	Phoenix	33.5	112.1	11	404.96	1963.46	404.96	4454.54
21	Salt Lake City	40.8	111.9	10	573.76	1931.68	573.76	5737.62
22	LA	34.1	118.4	18	71.11	2388.12	71.11	1280.04
23	SF	37.8	122.6	12	386.32	2661.52	386.32	4635.82
24	SD	32.8	117.1	10	154.65	2311.72	154.65	1546.47
25	Seattle	41.6	122.4	13	559.29	2658.20	559.29	7270.74
26								
27							Total	119676.47
28							Mean Dist	534.27
29								

Crouse Hospital Scheduling

The following example is a real problem from Crouse Hospital. The chief of surgery is trying to determine how to assign physicians attending (PA) the emergency room. He currently has 13 physicians.

1. The spreadsheet represents the different possible shifts in the emergency room (ER)



2. Historical data indicates the minimum number of physicians that must be on staff for each shift (row 27: PA Needed)
3. The # PA (B column) represents how many physicians should be assigned to each shift.
4. The PA working (row 25) counts how many physicians have been assigned to each time slot.
5. The total (cell B25) is the total number of physicians required.
6. Minimize the total number of physicians assigned subject to the constraint that each time slot has enough PA Working to cover the PA Needed (PA Working => PA Needed)
7. Don't forget the constraint that you need whole physicians, not fractional physicians