Homework #5

Due July 29 @ 11:59pm

Submission requirements

Upload a **single PDF file** of your IJulia notebook for this entire assigment. Clearly denote which question each section of your PDF corresponds to.

Problem 1 -- Nonconvex Quadratics

Suppose you have the constraint:

$$2x^2 + y^2 + 2z^2 - 3xy + xz - 4yz \leq 0 \qquad (1)$$

(a) Write constraint (1) in the standard form $v^T Q v \leq 0$ where Q is a symmetric matrix. What is Q and what is v?

Ans: vector $v = [x, y, z]$, symmetric matrix Q is $\begin{bmatrix} 2 & -1.5 & 0.5 \\ -1.5 & 1 & -2 \\ 0.5 & -2 & 2 \end{bmatrix}$

contstraint is $v^T Q v \leq 0$.

(b) This constraint is not convex (i.e., the set of points satisfying the constraint is not an ellipsoid). Explain why this is the case. *Hint*: You can perform an orthogonal decomposition of a symmetric matrix Q in Julia like this:

In [36]:
```
Q = [2 -1.5 0.5; -1.5 1 -2; 0.5 -2 2]

using LinearAlgebra
(L,U) = (eigvals(Q),eigvecs(Q)) # L is the vector of eigenvalues and U i
s orthogonal
U * Diagonal(L) * U' # this is equal to Q (as long as Q was symmetric to
begin with)

#take the square root of Diagonal(L) if all of its eigenvalues are non-n
egative
P = U * sqrt(Diagonal(L))
println(P)
```

```
DomainError with -0.8701601197116661:
sqrt will only return a complex result if called with a complex argumen
t. Try sqrt(Complex(x)).

Stacktrace:
 [1] throw_complex_domainerror(::Symbol, ::Float64) at ./math.jl:32
 [2] sqrt at ./math.jl:492 [inlined]
 [3] _broadcast_getindex_evalf at ./broadcast.jl:630 [inlined]
 [4] _broadcast_getindex at ./broadcast.jl:603 [inlined]
 [5] getindex at ./broadcast.jl:563 [inlined]
 [6] macro expansion at ./broadcast.jl:909 [inlined]
 [7] macro expansion at ./simdloop.jl:77 [inlined]
 [8] copyto! at ./broadcast.jl:908 [inlined]
 [9] copyto! at ./broadcast.jl:863 [inlined]
 [10] copy at ./broadcast.jl:839 [inlined]
 [11] materialize at ./broadcast.jl:819 [inlined]
 [12] sqrt(::Diagonal{Float64,Array{Float64,1}}) at /Users/julia/buildb
ot/worker/package_macos64/build/usr/share/julia/stdlib/v1.3/LinearAlgeb
ra/src/diagonal.jl:555
 [13] top-level scope at In[36]:6
```

Ans: The constraint is convex if and only if the matrix Q is symmetric and positive semidefinite. Since Q is not positive semidefinite for that we cannot take the sqaure root of it, the constraint is not convex.

(c) We can write constraint (1) in norm format as follows:

$$\|Av\|_2^2 - \|Bv\|_2^2 \leq 0 \qquad (2)$$

Find matrices A and B that make this constraint equivalent to (1).

Ans: $\|Av\|_2^2 - \|Bv\|_2^2 \leq 0$ is equivalent to $(v^T A^T Av) - (v^T B^T Bv) \leq 0$, hence equivalent to $v^T(A^T A - B^T B)v \leq 0$. Therefore, $Q = A^T A - B^T B$. Solve this using julia.

In [74]:
```
println(Diagonal(L))
#diagonize L and turn it into L_pos - L_neg
L_pos = [0.0 0.0 0.0; 0.0 1.518494119904299 0.0; 0.0 0.0 4.3516659998073
66]
L_neg = [-0.8701601197116661 0.0 0.0; 0.0 0 0.0; 0.0 0.0 0]

A = U * sqrt(L_pos)
B = U * sqrt(L_neg)
println("A is " ,A)
println("B is " ,B)
println(A*A'-B*B') # check: it should equals Q
```

```
[-0.8701601197116661 0.0 0.0; 0.0 1.518494119904299 0.0; 0.0 0.0 4.3516
65999807366]
A is [0.0 -0.9764959351583015 1.0685819183944123; -0.0 -0.0638199909800
2426 -1.2462660599533772; 0.0 0.7489170963964722 1.2870973511340034]
B is Complex{Float64}[0.0 - 0.30888772669073291im 0.0 + 0.0im 0.0 + 0.0i
m; -0.0 - 0.7464931904849552im -0.0 - 0.0im -0.0 - 0.0im; 0.0 - 0.46636
5102222593251im 0.0 + 0.0im 0.0 + 0.0im]
Complex{Float64}[1.9999999999999993 - 0.0im -1.5000000000000022 - 0.0im
0.4999999999999999 - 0.0im; -1.5000000000000022 - 0.0im 0.99999999999999
74 - 0.0im -1.9999999999999976 - 0.0im; 0.4999999999999999 - 0.0im -1.99
99999999999976 - 0.0im 2.0000000000000004 - 0.0im]
```

(d) Explain how to find (x, y, z) that satisfy the above constraint but make $2x^2 + y^2 + 2z^2$ arbitrarily large.

Ans: From the constraint, it is easy to notice that $2x^2 + y^2 + 2z^2 \leq 3xy - xz + 4yz$. Suppose $\frac{y}{\lambda_1} = \frac{z}{\lambda_2} = x$, then $2 + \lambda_1^2 + 2\lambda_2^2 \leq 3\lambda_1 - \lambda_2 + 4\lambda_1\lambda_2$ (2) by cancelling x^2. For arbitrarily large x, we need to show that there is always a feasible solution(λ_1, λ_2) such that (2) is satisfied.

In [80]:
```
using JuMP,Gurobi

m = Model(Gurobi.Optimizer)

@variable(m, λ[1:2])
@constraint(m, con, 2 + 2 + λ[1]^2 + 2λ[2]^2 <= 3λ[1] -λ[2] + 4λ[1]*λ[2])
@objective(m, Max, 2 + λ[1]^2 + 2λ[2]^2)


optimize!(m)
println(λ[1])
println(λ[2])
```

```
Academic license - for non-commercial use only
Academic license - for non-commercial use only
Gurobi Optimizer version 9.0.2 build v9.0.2rc0 (mac64)
Optimize a model with 0 rows, 2 columns and 0 nonzeros
Model fingerprint: 0xa0a87034
Model has 2 quadratic objective terms
Model has 1 quadratic constraint
Coefficient statistics:
  Matrix range     [0e+00, 0e+00]
  QMatrix range    [1e+00, 4e+00]
  QLMatrix range   [1e+00, 3e+00]
  Objective range  [0e+00, 0e+00]
  QObjective range [2e+00, 4e+00]
  Bounds range     [0e+00, 0e+00]
  RHS range        [0e+00, 0e+00]
  QRHS range       [4e+00, 4e+00]

Gurobi.GurobiError(10020, "Q matrix is not positive semi-definite (PS
D). Set NonConvex parameter to 2 to solve model.")

Stacktrace:
 [1] optimize at /Users/xuanzhouchen/.julia/packages/Gurobi/7YNJV/src/g
rb_solve.jl:7 [inlined]
 [2] optimize!(::Gurobi.Optimizer) at /Users/xuanzhouchen/.julia/packag
es/Gurobi/7YNJV/src/MOI_wrapper.jl:1920
 [3] optimize!(::MathOptInterface.Bridges.LazyBridgeOptimizer{Gurobi.Op
timizer}) at /Users/xuanzhouchen/.julia/packages/MathOptInterface/bygN
7/src/Bridges/bridge_optimizer.jl:239
 [4] optimize!(::MathOptInterface.Utilities.CachingOptimizer{MathOptInt
erface.AbstractOptimizer,MathOptInterface.Utilities.UniversalFallback{M
athOptInterface.Utilities.Model{Float64}}}) at /Users/xuanzhouchen/.jul
ia/packages/MathOptInterface/bygN7/src/Utilities/cachingoptimizer.jl:18
9
 [5] #optimize!#97(::Bool, ::Bool, ::Base.Iterators.Pairs{Union{},Union
{},Tuple{},NamedTuple{(),Tuple{}}}, ::typeof(optimize!), ::Model, ::Not
hing) at /Users/xuanzhouchen/.julia/packages/JuMP/YXK4e/src/optimizer_i
nterface.jl:131
 [6] optimize! at /Users/xuanzhouchen/.julia/packages/JuMP/YXK4e/src/op
timizer_interface.jl:107 [inlined] (repeats 2 times)
 [7] top-level scope at In[80]:8
```

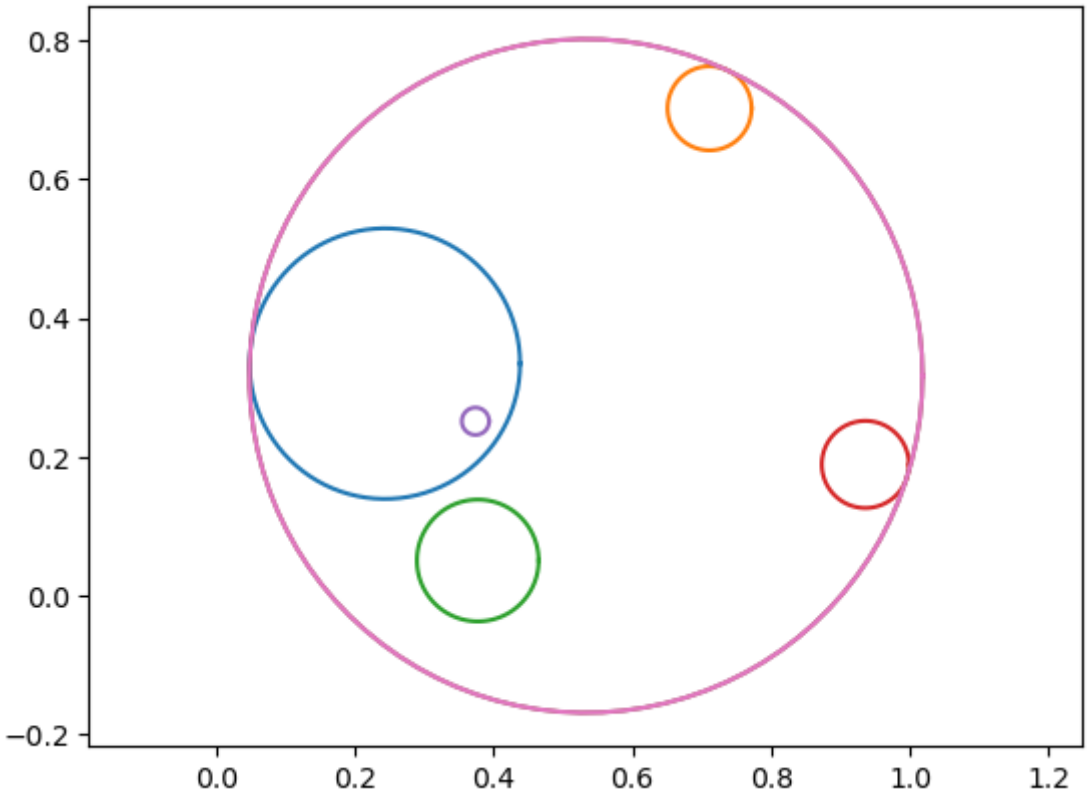
Problem 2 -- Circles within Circles

Formulate a convex program to solve the minimum enclosing ball (MEB) problem. You need to determine the center z of a ball, and its radius d so that each of the circles centered at c_i with radius r_i are enclosed in your generated circle. Your task is to find the cirle of smallest radius that encircles all the other circles.

You can use the following code to start building your model. Start with $n = 5$ circles to help with debugging, but display the output of your model for both $n = 5$ and a very large value of n, such as $n = 200$.

In [52]:
```
n = 5
using Random, PyPlot
r = [rand()*0.2 for i in 1:n] # randomly generate n radii with length be
tween 0 and 0.2
# randomly generate n points with (x,y) coords between 0 and 1
c = [rand() for i in 1:n, j in 1:2]

t = range(0,stop=2*pi,length=100) # parameter that traverses the circle

# for each of the n points
for i in 1:n
    # plot circle radius r[i] with center (x[i],y[i])
    plot( c[i,1] .+ r[i]*cos.(t), c[i,2] .+ r[i]*sin.(t))
end

axis("equal"); # make x and y scales equal

#build model
using Pkg
Pkg.add("Ipopt")
using JuMP,Ipopt

m = Model(Ipopt.Optimizer)

@variable(m, x[1:2])# variable for coordinates of circle center
@variable(m, d>=0)# variable representing radius of the circle

@NLobjective(m, Min, pi*(d^2))# we want to minimize the area

@NLconstraint(m, constr[i in 1:n], (x[1]-c[i,1])^2 + (x[2]-c[i,2])^2 <=
(d-r[i])^2 )

optimize!(m)
t=range(0,stop=2pi,length=100)# parameter that traverses the circle
# plot circle radius d with center (x1,x2); (orange circe)
plot( value(x[1]).+ value(d)*cos.(t), value(x[2]).+ value(d)*sin.(t));
```

```
Resolving package versions...
```



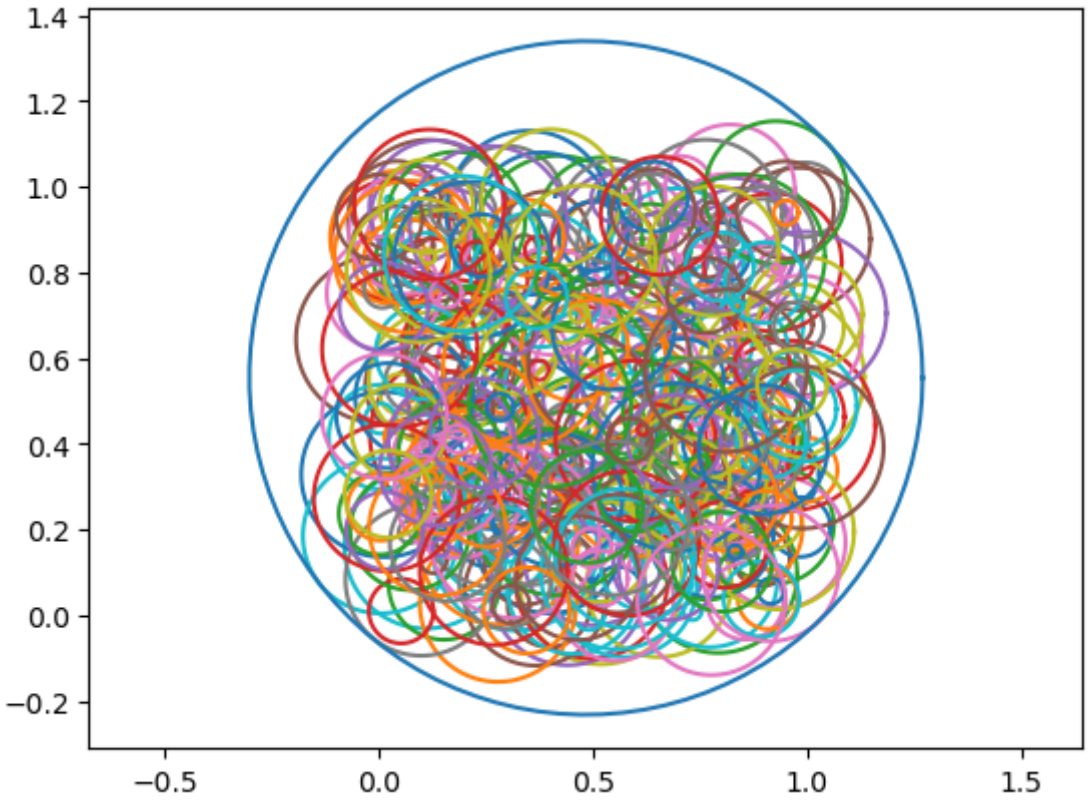
```
  Updating `~/.julia/environments/v1.3/Project.toml`
 [no changes]
  Updating `~/.julia/environments/v1.3/Manifest.toml`
 [no changes]
This is Ipopt version 3.13.2, running with linear solver mumps.
NOTE: Other linear solvers might be more efficient (see Ipopt documenta
tion).

Number of nonzeros in equality constraint Jacobian...:        0
Number of nonzeros in inequality constraint Jacobian.:       15
Number of nonzeros in Lagrangian Hessian.............:       16

Total number of variables............................:        3
                     variables with only lower bounds:        1
                variables with lower and upper bounds:        0
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:        5
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:        5

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
   0  3.1415864e-04 9.97e-01 1.24e+00  -1.0 0.00e+00    -  0.00e+00 0.0
0e+00   0
   1  6.5624396e-02 2.54e-01 2.03e+01  -1.0 4.67e-01    -  5.80e-02 1.0
0e+00f  1
   2  8.3360364e-02 2.15e-01 1.74e+01  -1.0 3.37e-01   2.0 1.00e+00 1.8
9e-01h  1
   3  3.5227950e-01 8.64e-02 1.43e+01  -1.0 3.49e-01   1.5 1.00e+00 4.9
2e-01h  1
   4  1.1661677e+00 0.00e+00 3.40e+01  -1.0 2.74e-01    -  3.04e-01 1.0
0e+00f  1
   5  1.2527991e+00 0.00e+00 6.29e+00  -1.0 1.35e-01   1.0 1.00e+00 1.0
0e+00h  1
   6  1.0239001e+00 0.00e+00 2.31e-01  -1.0 1.02e-01    -  1.00e+00 1.0
0e+00f  1
   7  7.9730222e-01 0.00e+00 1.25e-02  -1.7 9.75e-02    -  1.00e+00 1.0
0e+00h  1
   8  7.4819585e-01 0.00e+00 6.33e-02  -2.5 1.80e-02    -  1.00e+00 8.7
6e-01h  1
   9  7.4947098e-01 0.00e+00 9.64e-04  -2.5 6.15e-03    -  1.00e+00 1.0
0e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
  10  7.4103690e-01 0.00e+00 3.28e-04  -3.8 5.72e-03    -  1.00e+00 1.0
0e+00h  1
  11  7.4055185e-01 6.64e-07 1.04e-05  -5.7 1.16e-03    -  1.00e+00 1.0
0e+00h  1
  12  7.4053863e-01 7.29e-10 8.25e-09  -8.6 4.06e-05    -  1.00e+00 1.0
0e+00h  1
  13  7.4053862e-01 0.00e+00 1.00e-14  -9.0 4.52e-08    -  1.00e+00 1.0
0e+00h  1

Number of Iterations....: 13
```

```
                                   (scaled)                 (unscaled)
Objective...............:   7.4053861765729734e-01    7.4053861765729734e-01
Dual infeasibility......:   1.0002480044416343e-14    1.0002480044416343e-14
Constraint violation....:   0.0000000000000000e+00    0.0000000000000000e+00
Complementarity.........:   9.0911049503938839e-10    9.0911049503938839e-10
Overall NLP error.......:   9.0911049503938839e-10    9.0911049503938839e-10


Number of objective function evaluations             = 14
Number of objective gradient evaluations             = 14
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 14
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 14
Number of Lagrangian Hessian evaluations             = 13
Total CPU secs in IPOPT (w/o function evaluations)   =      0.011
Total CPU secs in NLP function evaluations           =      0.000

EXIT: Optimal Solution Found.
This is Ipopt version 3.13.2, running with linear solver mumps.
NOTE: Other linear solvers might be more efficient (see Ipopt documenta
tion).

Number of nonzeros in equality constraint Jacobian...:        0
Number of nonzeros in inequality constraint Jacobian.:       15
Number of nonzeros in Lagrangian Hessian.............:       16

Total number of variables............................:        3
                     variables with only lower bounds:        1
                variables with lower and upper bounds:        0
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:        5
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:        5

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
   0  3.1415864e-04 9.97e-01 1.24e+00  -1.0 0.00e+00    -  0.00e+00 0.0
0e+00   0
   1  6.5624396e-02 2.54e-01 2.03e+01  -1.0 4.67e-01    -  5.80e-02 1.0
0e+00f  1
   2  8.3360364e-02 2.15e-01 1.74e+01  -1.0 3.37e-01   2.0 1.00e+00 1.8
9e-01h  1
   3  3.5227950e-01 8.64e-02 1.43e+01  -1.0 3.49e-01   1.5 1.00e+00 4.9
2e-01h  1
   4  1.1661677e+00 0.00e+00 3.40e+01  -1.0 2.74e-01    -  3.04e-01 1.0
0e+00f  1
   5  1.2527991e+00 0.00e+00 6.29e+00  -1.0 1.35e-01   1.0 1.00e+00 1.0
0e+00h  1
   6  1.0239001e+00 0.00e+00 2.31e-01  -1.0 1.02e-01    -  1.00e+00 1.0
```

```
0e+00f  1
   7  7.9730222e-01 0.00e+00 1.25e-02  -1.7 9.75e-02    -  1.00e+00 1.0
0e+00h  1
   8  7.4819585e-01 0.00e+00 6.33e-02  -2.5 1.80e-02    -  1.00e+00 8.7
6e-01h  1
   9  7.4947098e-01 0.00e+00 9.64e-04  -2.5 6.15e-03    -  1.00e+00 1.0
0e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
  10  7.4103690e-01 0.00e+00 3.28e-04  -3.8 5.72e-03    -  1.00e+00 1.0
0e+00h  1
  11  7.4055185e-01 6.64e-07 1.04e-05  -5.7 1.16e-03    -  1.00e+00 1.0
0e+00h  1
  12  7.4053863e-01 7.29e-10 8.25e-09  -8.6 4.06e-05    -  1.00e+00 1.0
0e+00h  1
  13  7.4053862e-01 0.00e+00 1.00e-14  -9.0 4.52e-08    -  1.00e+00 1.0
0e+00h  1

Number of Iterations....: 13

                                   (scaled)                 (unscaled)
Objective...............:   7.4053861765729734e-01    7.405386176572973
4e-01
Dual infeasibility......:   1.0002480044416343e-14    1.000248004441634
3e-14
Constraint violation....:   0.0000000000000000e+00    0.000000000000000
0e+00
Complementarity.........:   9.0911049503938839e-10    9.091104950393883
9e-10
Overall NLP error.......:   9.0911049503938839e-10    9.091104950393883
9e-10


Number of objective function evaluations             = 14
Number of objective gradient evaluations             = 14
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 14
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 14
Number of Lagrangian Hessian evaluations             = 13
Total CPU secs in IPOPT (w/o function evaluations)   =      0.011
Total CPU secs in NLP function evaluations           =      0.000

EXIT: Optimal Solution Found.
```

In [58]:
```
n = 400
using Random, PyPlot
r = [rand()*0.2 for i in 1:n] # randomly generate n radii with length be
tween 0 and 0.2
# randomly generate n points with (x,y) coords between 0 and 1
c = [rand() for i in 1:n, j in 1:2]

t = range(0,stop=2*pi,length=100) # parameter that traverses the circle

# for each of the n points
for i in 1:n
    # plot circle radius r[i] with center (x[i],y[i])
    plot( c[i,1] .+ r[i]*cos.(t), c[i,2] .+ r[i]*sin.(t))
end

axis("equal"); # make x and y scales equal

#build model
using Pkg
Pkg.add("Ipopt")
using JuMP,Ipopt

m = Model(Ipopt.Optimizer)

@variable(m, x[1:2])# variable for coordinates of circle center
@variable(m, d>=0)# variable representing radius of the circle

@NLobjective(m, Min, pi*(d^2))# we want to minimize the area

@NLconstraint(m, constr[i in 1:n], (x[1]-c[i,1])^2 + (x[2]-c[i,2])^2 <=
(d-r[i])^2 )

optimize!(m)
t=range(0,stop=2pi,length=100)# parameter that traverses the circle
# plot circle radius d with center (x1,x2);
plot( value(x[1]).+ value(d)*cos.(t), value(x[2]).+ value(d)*sin.(t));
```

```
Resolving package versions...
```



```
  Updating `~/.julia/environments/v1.3/Project.toml`
 [no changes]
  Updating `~/.julia/environments/v1.3/Manifest.toml`
 [no changes]
This is Ipopt version 3.13.2, running with linear solver mumps.
NOTE: Other linear solvers might be more efficient (see Ipopt documenta
tion).

Number of nonzeros in equality constraint Jacobian...:        0
Number of nonzeros in inequality constraint Jacobian.:     1200
Number of nonzeros in Lagrangian Hessian.............:     1201

Total number of variables............................:        3
                     variables with only lower bounds:        1
                variables with lower and upper bounds:        0
                     variables with only upper bounds:        0
Total number of equality constraints.................:        0
Total number of inequality constraints...............:      400
        inequality constraints with only lower bounds:        0
   inequality constraints with lower and upper bounds:        0
        inequality constraints with only upper bounds:      400

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
   0  3.1415864e-04 1.93e+00 1.66e+00  -1.0 0.00e+00    -  0.00e+00 0.0
0e+00   0
   1  1.8715516e-03 1.87e+00 1.90e+00  -1.0 4.46e-01    -  2.88e-02 3.3
2e-02f  1
   2  2.7477677e-03 1.86e+00 4.15e+01  -1.0 6.11e-01    -  4.82e-02 8.4
6e-03f  1
   3  1.1977085e-01 1.15e+00 2.10e+02  -1.0 5.15e-01    -  8.71e-02 4.0
7e-01f  1
   4  3.2543085e-01 1.02e+00 1.91e+02  -1.0 1.38e+00    -  1.40e-01 9.1
8e-02f  1
   5  3.5181592e-01 9.94e-01 1.87e+02  -1.0 6.20e-01    -  1.06e-01 2.0
6e-02h  1
   6  3.2763826e+00 0.00e+00 3.83e+02  -1.0 6.87e-01    -  2.58e-01 1.0
0e+00f  1
   7  3.0836917e+00 0.00e+00 3.14e+02  -1.0 3.50e-01    -  1.07e-01 1.8
4e-01h  1
   8  2.8735763e+00 2.58e-02 7.74e+01  -1.0 4.16e-01    -  8.92e-02 1.0
0e+00f  1
   9  4.1521286e+00 0.00e+00 1.83e+01  -1.0 4.25e-01    -  4.51e-01 1.0
0e+00f  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
  10  6.3270018e+00 0.00e+00 7.76e+00  -1.0 7.19e-01    -  3.31e-01 1.0
0e+00f  1
  11  8.5481281e+00 0.00e+00 6.54e-01  -1.0 6.59e-01    -  4.68e-01 1.0
0e+00f  1
  12  6.8590134e+00 0.00e+00 2.17e-01  -1.7 4.15e-01    -  6.18e-01 1.0
0e+00h  1
  13  7.1030457e+00 0.00e+00 1.42e-02  -1.7 4.25e-01    -  1.00e+00 1.0
0e+00h  1
  14  3.2323501e+00 0.00e+00 8.83e-02  -2.5 1.68e+00    -  1.00e+00 9.1
6e-01h  1
  15  2.3408544e+00 0.00e+00 4.33e-02  -2.5 1.51e-01    -  1.00e+00 1.0
```

```
0e+00h  1
  16  2.2059698e+00 0.00e+00 2.16e-03  -2.5 4.56e-02    -  1.00e+00 1.0
0e+00h  1
  17  1.9865563e+00 0.00e+00 2.68e-01  -3.8 2.97e-01    -  1.00e+00 2.6
9e-01f  1
  18  1.9607147e+00 0.00e+00 1.16e+00  -3.8 8.96e-02    -  1.00e+00 1.5
7e-01h  1
  19  1.9423307e+00 4.22e-04 2.21e-01  -3.8 3.29e-02    -  7.20e-01 1.0
0e+00h  1
iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alp
ha_pr  ls
  20  1.9418250e+00 1.16e-04 5.27e-03  -3.8 1.01e-02    -  1.00e+00 8.4
1e-01h  1
  21  1.9426394e+00 0.00e+00 2.05e-06  -3.8 1.80e-03    -  1.00e+00 1.0
0e+00h  1
  22  1.9419956e+00 1.75e-06 3.42e-02  -5.7 2.27e-03    -  9.93e-01 7.8
7e-01h  1
  23  1.9419559e+00 0.00e+00 1.26e-07  -5.7 3.44e-05    -  1.00e+00 1.0
0e+00h  1
  24  1.9419504e+00 3.39e-11 9.88e-06  -8.6 9.45e-06    -  1.00e+00 9.9
7e-01h  1
  25  1.9419504e+00 0.00e+00 2.86e-14  -8.6 1.38e-08    -  1.00e+00 1.0
0e+00f  1

Number of Iterations....: 25

                                   (scaled)                 (unscaled)
Objective...............:   1.9419504140556154e+00    1.941950414055615
4e+00
Dual infeasibility......:   2.8634332171460145e-14    2.863433217146014
5e-14
Constraint violation....:   0.0000000000000000e+00    0.000000000000000
0e+00
Complementarity.........:   2.5059158220662008e-09    2.505915822066200
8e-09
Overall NLP error.......:   2.5059158220662008e-09    2.505915822066200
8e-09


Number of objective function evaluations             = 26
Number of objective gradient evaluations             = 26
Number of equality constraint evaluations            = 0
Number of inequality constraint evaluations          = 26
Number of equality constraint Jacobian evaluations   = 0
Number of inequality constraint Jacobian evaluations = 26
Number of Lagrangian Hessian evaluations             = 25
Total CPU secs in IPOPT (w/o function evaluations)   =      0.083
Total CPU secs in NLP function evaluations           =      0.012

EXIT: Optimal Solution Found.
```

In [ ]: