In [32]:
```
using Hyperopt

f(x,a,b=true;c=10) = sum(@. x + (a-3)^2 + (b ? 10 : 20) + (c-100)^2) # Function to minimize

# Main macro. The first argument to the for loop is always interpreted as the number of iterations
ho = @hyperopt for i=50,
            sampler = RandomSampler(), # This is default if none provided
            a = LinRange(1,5,1000),
            b = [true, false],
            c = exp10.(LinRange(-1,3,1000))
   print(i, "\t", a, "\t", b, "\t", c, "   \t")
   x = 100
   @show f(x,a,b,c=c)
end
```

```
1       3.7027027027027026      false   397.7403024058037       f(x, a, b, c = c) = 88769.7814677878
2       3.2622622622622623      true    13.005112521734086      f(x, a, b, c = c) = 7678.179228850353
3       3.3263263263263263      true    0.3165924111983522      f(x, a, b, c = c) = 10046.88823738641
4       3.942942942942943       false   1.9646866461804455      f(x, a, b, c = c) = 9731.811805775238
5       3.890890890890891       false   6.9477125484602364      f(x, a, b, c = c) = 8779.521886543458
6       1.3523523523523524      true    0.13936192742241424     f(x, a, b, c = c) = 10084.86177903313
7       4.007007007007007       true    24.343688735431105      f(x, a, b, c = c) = 5834.8914972734965
8       4.051051051051051       false   963.7934799615786       f(x, a, b, c = c) = 746260.280732416
9       1.4084084084084085      false   78.50456200204509       f(x, a, b, c = c) = 584.5870185183488
10      2.5495495495495497      true    188.48434090337955      f(x, a, b, c = c) = 7939.681490713798
11      1.7327327327327327      false   469.5390010680058       f(x, a, b, c = c) = 136680.67927666628
12      1.924924924924925       true    0.11067360180959737     f(x, a, b, c = c) = 10089.033314701266
13      1.4404404404404405      false   44.32478591240396       f(x, a, b, c = c) = 3222.1616897194663
14      2.2812812812812813      false   56.85317913873753       f(x, a, b, c = c) = 1982.164707030511
15      1.028028028028028       false   204.79120966650854      f(x, a, b, c = c) = 11105.086296828395
16      2.1771771771771773      true    0.7886728615614151      f(x, a, b, c = c) = 9953.56416996804
17      4.671671671671672       true    26.4498018242772        f(x, a, b, c = c) = 5522.426137865967
18      4.067067067067067       true    218.4436071149426       f(x, a, b, c = c) = 14140.026698524503
19      4.651651651651652       false   0.4252346334528682      f(x, a, b, c = c) = 10037.861850981317
20      3.7387387387387387      true    0.3737425742391064      f(x, a, b, c = c) = 10035.936903588092
21      2.4734734734734736      false   24.120282076180068      f(x, a, b, c = c) = 5878.008822381616
22      4.563563563563564       true    0.3737425742391064      f(x, a, b, c = c) = 10037.835899681282
23      1.5645645645645647      true    0.3669142378402493      f(x, a, b, c = c) = 10038.812253379185
24      2.413413413413413       true    0.14459729217920195     f(x, a, b, c = c) = 10081.445533764629
25      2.641641641641642       false   346.3694177371734       f(x, a, b, c = c) = 60818.01841686687
26      3.4864864864864864      false   1.6188596901781984      f(x, a, b, c = c) = 9799.085437762378
27      5.0     true    0.4749814803228501      f(x, a, b, c = c) = 10019.229311342078
28      4.971971971971972       true    0.8728526623848377      f(x, a, b, c = c) = 9940.080012751507
29      1.6486486486486487      false   141.6286616299199       f(x, a, b, c = c) = 1854.7716195731637
30      2.7177177177177176      true    1.3339056900390587      f(x, a, b, c = c) = 9845.077849668998
31      3.894894894894895       true    0.27570233256095833     f(x, a, b, c = c) = 10055.736382136898
32      3.958958958958959       false   1.46273335620113        f(x, a, b, c = c) = 9830.512519916087
33      3.8628628628628627      true    210.5345242766706       f(x, a, b, c = c) = 12328.625589389989
34      3.234234234234234       true    519.6557243827663       f(x, a, b, c = c) = 176220.98187290077
35      2.6176176176176176      false   1.0115911122238297      f(x, a, b, c = c) = 9918.851310419923
36      2.88988988988989        false   839.3129498166361       f(x, a, b, c = c) = 546703.6498908127
37      4.1391391391391394      true    13.005112521734086      f(x, a, b, c = c) = 7679.408085334465
38      4.003003003003003       false   14.661086840469844      f(x, a, b, c = c) = 7403.736114273862
39      3.1781781781781784      true    710.9709432312432       f(x, a, b, c = c) = 373395.5252203381
40      2.2212212212212212      false   18.29204504846291       f(x, a, b, c = c) = 6796.796398718686
41      3.002002002002002       true    0.3195247505759212      f(x, a, b, c = c) = 10046.197149959058
42      2.921921921921922       false   390.473523688556        f(x, a, b, c = c) = 84494.87406023237
43      2.1291291291291294      true    46.41588833612777       f(x, a, b, c = c) = 2982.0154388800593
44      1.1361361361361362      true    0.8490115201088749      f(x, a, b, c = c) = 9941.38655592462
45      3.6346346346346348      true    0.34082585474234517     f(x, a, b, c = c) = 10042.353752434268
46      4.123123123123123       false   119.97177354358843      f(x, a, b, c = c) = 520.1331440260724
47      4.771771771771772       true    0.8183006815867392      f(x, a, b, c = c) = 9950.148654899385
48      4.307307307307307       false   0.33769803108250906     f(x, a, b, c = c) = 10054.283486139435
49      4.495495495495495       true    191.99206655932846      f(x, a, b, c = c) = 8574.776816632966
50      2.841841841841842       false   19.15500555573528       f(x, a, b, c = c) = 6655.938140696187
```

Out[32]:
```
Hyperoptimizer
  iterations: Int64 50
  params: Tuple{Symbol, Symbol, Symbol}
  candidates: Tuple{LinRange{Float64}, Vector{Bool}, Vector{Float64}}
  history: Array{Any}((50,))
  results: Array{Any}((50,))
  sampler: RandomSampler RandomSampler()
```

In [33]:
```
printmin(ho)
```

```
a = 4.123123123123123
b = false
c = 119.97177354358843
```

In [34]:
```
ho = Hyperoptimizer(10, a = LinRange(1,2,50), b = [true, false], c = randn(100))
for (i,a,b,c) in ho
    println(i, "\t", a, "\t", b, "\t", c)
end
```

```
1       1.3265306122448979      false   1.5078794067521035
2       1.6122448979591837      true    1.0976461649059748
3       1.1020408163265305      false   0.6041764103214882
4       1.510204081632653       false   -0.6691233108564895
5       1.3265306122448979      true    -1.66042037919107
6       1.0816326530612246      false   0.9874537165843437
7       1.7755102040816326      true    -0.9354565543517442
8       1.5918367346938775      true    1.0976461649059748
9       1.5510204081632653      false   0.6041764103214882
10      1.1224489795918366      true    -0.8900301494615032
```

In [40]:
```
hob = @hyperopt for i=100, sampler = LHSampler(),
                           a = LinRange(1,5,100),
                           b = repeat([true, false],50),
                           c = exp10.(LinRange(-1,3,100))
    f(a,b,c=c)
end
```

Out[40]:
```
Hyperoptimizer
  iterations: Int64 100
  params: Tuple{Symbol, Symbol, Symbol}
  candidates: Tuple{LinRange{Float64}, Vector{Bool}, Vector{Float64}}
  history: Array{Any}((100,))
  results: Array{Any}((100,))
  sampler: LHSampler
```

In [41]:
```
hob = @hyperopt for i=100,
                    sampler=CLHSampler(dims=[Continuous(), Categorical(2), Continuous()]),
                    a = LinRange(1,5,100),
                    b = [true, false],
                    c = exp10.(LinRange(-1,3,100))
    f(a,b,c=c)
end
```

Out[41]:
```
Hyperoptimizer
  iterations: Int64 100
  params: Tuple{Symbol, Symbol, Symbol}
  candidates: Tuple{LinRange{Float64}, Vector{Bool}, Vector{Float64}}
  history: Array{Any}((100,))
  results: Array{Any}((100,))
  sampler: CLHSampler
```

In [ ]:
```
ho = @hyperopt for i=18, sampler=Hyperband(R=50, η=3, inner=RandomSampler()), a = LinRange(1,5,1800), c = exp10.(LinRange(
    if state === nothing # Query if state is initialized
        res = optimize(resources=i, a, b) # if state is uninitialized, start a new optimization using the selected hyper para
    else
        res = optimize(resources=i, state=state) # If state has a value, continue the optimization from the state
    end
    minimum(res), get_state(res) # return the minimum value and a state from which to continue the optimization
end
```

In [ ]:
```
using Optim
f(a;c=10) = sum(@. 100 + (a-3)^2 + (c-100)^2)
hohb = @hyperopt for i=18, sampler=Hyperband(R=50, η=3, inner=RandomSampler()), a = LinRange(1,5,1800), c = exp10.(LinRang
    if !(state === nothing)
        a,c = state
    end
    res = Optim.optimize(x->f(x[1],c=x[2]), [a,c], SimulatedAnnealing(), Optim.Options(f_calls_limit=i))
    Optim.minimum(res), Optim.minimizer(res)
end
#plot(hohb)
```

In [ ]: