

# validation\_runs

March 26, 2019

```
In [1]: import ingestor, modeller, fitter
import numpy as np
from matplotlib import pyplot as plt

In [2]: plt.rc('text', usetex=True)
plt.rc('font', family='serif')

In [3]: from importlib import reload

        reload(ingestor)
        reload(modeller)
        reload(fitter)

Out[3]: <module 'fitter' from '/media/dwu402/Data/wrap-mad/fitter.py'>

In [4]: context = ingestor.initialise_context()
        ingestor.read_run_file(context, "runs/mouse4.3.run")

In [5]: model = modeller.Model(context)

In [6]: solver = fitter.Fitter()
        solver.construct_objectives(context, model)

In [7]: solver.construct_problems()
        print(solver.solutions)

{}

In [8]: for rhoi in np.logspace(1, 5, num=41):
        solver.solve(rhoi)

In [9]: solver.solutions

Out[9]: {'10.0': [      fun: 0.11659342994557494
                hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
                jac: array([ 0.0016526 ,  0.00049605,  0.00521841,  0.00294731, -0.005592  ,
                -0.00252685,  0.00212057, -0.00043477,  0.05141717])
                message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
```

```

nfev: 188
nit: 123
status: 0
success: True
x: array([ 0.          ,  0.          ,  0.          , 91.84004789, 48.38994169,
          11.39322897, 13.62398031,  2.82618728,  0.          ])],
'12.589254117941675': [      fun: 0.12689545937070779
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([ 1.62262212e-04,  5.72517405e-04,  9.12655476e-06, -2.70050049e-07,
           -4.32936762e-06,  2.03886496e-06, -6.54391719e-06,  1.13277878e-05,
           1.24321181e-02])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 107
nit: 78
status: 0
success: True
x: array([0.00000000e+00, 0.00000000e+00, 9.42696516e-04, 1.00000000e+02,
          7.63291604e+00, 4.54095841e+01, 1.28293000e+01, 3.12298983e+00,
          0.00000000e+00])),
'15.848931924611133': [      fun: 0.19477400688114158
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([ 0.05675171, -0.09352872, -0.02062169,  0.03072648, -0.15088322,
           -0.02125112,  0.01324966, -0.00789147,  0.06299941])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 90
nit: 18
status: 0
success: True
x: array([0.75301797, 0.24566167, 0.34902971, 1.76372751, 0.9960068 ,
          1.35863793, 2.52376518, 1.65392309, 0.42539134])),
'19.952623149688797': [      fun: 0.20639630737536063
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([-0.00542878, -0.00067046,  0.00281855,  0.01028904, -0.00620384,
           0.00036287,  0.00750931,  0.00136792, -0.00022133])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 84
nit: 19
status: 0
success: True
x: array([0.84699107, 0.80595483, 2.87032139, 0.          , 0.96351507,
          3.20544297, 3.08109443, 1.84083944, 0.59836558])),
'25.118864315095795': [      fun: 0.23737376304820434
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([ 0.03880436, -0.04491561, -0.06293356, -0.06646954,  0.17452177,
           -0.03366186,  0.0386899 ,  0.00112117,  0.02214604])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 75
nit: 10

```

```

    status: 0
    success: True
        x: array([0.73491543, 0.4900263 , 1.42053168, 0.8713086 , 1.10262451,
        2.34198225, 2.03418136, 1.18656149, 0.48929575]))],
'31.622776601683793': [      fun: 0.1630913674460301
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 4.59312845e-04,  2.05266258e-04,  1.37222607e-03, -6.84774049e-10,
        2.98024909e-06, -6.11628371e-06,  8.24236216e-06,  6.60723256e-07,
        9.23172154e-06])
    message: b'CONVERGENCE: NORM_OF_PROJECTED_GRADIENT_<=_PGTOL'
        nfev: 52
        nit: 47
    status: 0
    success: True
        x: array([ 0.          ,  0.          ,  0.          , 30.54075164,
        25.21064604, 100.          ,  0.22057362,  4.22025543,
        52.83803409]))],
'39.810717055349734': [      fun: 0.2463391632695019
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.00549282,  0.01270791, -0.00714957,  0.00842344,  0.02267731,
        -0.04003929,  0.0355226 ,  0.00236757,  0.05134891])
    message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 27
        nit: 10
    status: 0
    success: True
        x: array([0.73996834, 0.69792425, 1.55541149, 1.15550605, 1.09082566,
        2.29960709, 1.79615439, 1.07109252, 0.80066941]))],
'50.11872336272725': [      fun: 0.20289897751087396
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 0.02167905, -0.00390307,  0.00081937,  0.00093899,  0.00443873,
        -0.00273187, -0.00498159, -0.00042141,  0.0016666 ] )
    message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 84
        nit: 29
    status: 0
    success: True
        x: array([0.51258032, 1.94613531, 0.86480045, 4.56954646, 2.48026494,
        8.61181544, 0.55771493, 2.11786761, 0.          ])],
'63.09573444801933': [      fun: 0.5419234560749957
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.01428739,  0.00824968, -0.06290991,  0.34377816, -0.67534343,
        0.15808217, -0.35180032, -0.00957227, -0.35570499])
    message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 13
        nit: 1
    status: 0
    success: True

```

```

        x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
1.e-03])),
'79.43282347242814': [      fun: 0.12598178945124935
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([ 0.00103219,  0.06819242,  0.02897217,  0.05903362, -0.15308502,
0.13131372, -0.07265371,  0.00718017,  0.04405511])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 109
nit: 38
status: 0
success: True
x: array([1.62719304, 0.91534716, 3.1416983 , 1.4411207 , 1.03449222,
0.2759238 , 1.31930648, 0.77852809, 1.82344925])),
'100.0': [      fun: 0.6479341257361756
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([-0.04850016,  0.05377976, -0.15167413,  0.68458465, -1.32184264,
0.66627862, -0.93547964, -0.00706081, -0.24710592])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 66
nit: 1
status: 0
success: True
x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
1.e-03])),
'125.89254117941675': [      fun: 0.3827709192903993
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([-0.05665304, -0.08264215, -0.01587849,  0.01706979, -0.10471139,
-0.0440889 , -0.12652397, -0.03870491, -0.16210815])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 11
nit: 3
status: 0
success: True
x: array([0.01138727, 1.07772742, 1.08243327, 2.01651479, 0.75431349,
1.15699975, 0.03198216, 2.67032713, 0.07190011])),
'158.48931924611142': [      fun: 0.817266704839094
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
jac: array([-0.09001434,  0.11581746, -0.26139384,  1.19016256, -2.29833872,
0.99191573, -1.35285029,  0.00444554, -0.27874477])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
nfev: 14
nit: 1
status: 0
success: True
x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
1.e-03])),
'199.52623149688787': [      fun: 0.2616804977991779
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>

```

```

        jac: array([-0.01930609,  0.03866896,  0.00279795,  0.02247842, -0.04092318,
        0.00160024, -0.00457997, -0.00424377, -0.04699521])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 22
    nit: 13
    status: 0
    success: True
        x: array([0.          , 0.27240016, 7.90860973, 1.16622908, 1.32693392,
        4.56491892, 5.03591439, 4.11573213, 0.07441785])),
'251.18864315095823': [    fun: 1.0612410010972242
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.18543964,  0.2456274 , -0.43151259,  1.79957683, -3.36889487,
        1.52929491, -2.11236963,  0.03404403, -0.29077811])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 74
    nit: 1
    status: 0
    success: True
        x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
        1.e-03])),
'316.22776601683796': [    fun: 0.4994101109968719
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.43450731,  0.39503702, -0.32917993, -0.46137724,  1.36344875,
        -0.57081673,  0.39337722, -0.01169313, -0.1339704 ])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 38
    nit: 7
    status: 0
    success: True
        x: array([0.38249871, 0.83486147, 1.09535759, 1.73489142, 1.13014564,
        1.35886416, 1.34495946, 0.91675589, 0.67420274])),
'398.1071705534973': [    fun: 1.432831518795043
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.30654314,  0.38967655, -0.58653225,  2.41131875, -4.27917557,
        2.10629894, -2.84910659,  0.02971691, -0.32387023])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 33
    nit: 1
    status: 0
    success: True
        x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
        1.e-03])),
'501.18723362727246': [    fun: 0.6391416543014414
    hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 0.10479282,  0.18512808, -0.33213641, -0.79532685,  1.72778234,
        -0.61402372,  0.20663913, -0.02547235, -0.20240736])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 83

```

```

        nit: 9
        status: 0
        success: True
        x: array([0.62665261, 0.75061607, 1.40470487, 1.69666341, 1.12130413,
        1.17732592, 1.3244632 , 0.99528699, 0.54487028])),
'630.957344480193': [      fun: 1.8189265019018015
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([-0.20537134,  0.27773471, -0.53644538,  4.90788381, -8.80781801,
        2.61731892, -3.73406678, -0.2678318 , -0.62426635])
        message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 10
        nit: 1
        status: 0
        success: True
        x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
        1.e-03])),
'794.3282347242822': [      fun: 0.2755963138265205
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 0.05870534, -0.02236198, -0.00735068,  0.02193022,  0.00532327,
        0.01241255, -0.02739   , -0.00154544, -0.00960985])
        message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 51
        nit: 13
        status: 0
        success: True
        x: array([0.49859457, 0.38583349, 4.86915203, 1.42160165, 1.06307871,
        2.96436757, 0.9244606 , 0.96056498, 0.94681649])),
'1000.0': [      fun: 0.23981826600119177
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 5.15405511e-02, -4.06749435e-02, -8.01130416e-05, -2.60335636e-04,
        3.58330561e-04, -1.53280235e-04,  1.07673377e-04,  1.91200916e-03,
        7.98767580e-03])
        message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 99
        nit: 47
        status: 0
        success: True
        x: array([0.00000000e+00, 3.04819818e-02, 7.52025038e+01, 3.85900057e-02,
        1.73261135e+01, 7.56482077e+01, 6.70781373e+01, 3.23924793e-01,
        8.38832047e-01])),
'1258.9254117941675': [      fun: 0.2897049351009588
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
        jac: array([ 0.04763707, -0.01033337,  0.00052549, -0.00651193, -0.0060979 ,
        0.00297254, -0.00921433,  0.00191381, -0.01056027])
        message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
        nfev: 41
        nit: 21
        status: 0

```

```

success: True
      x: array([0.21979035, 0.42486619, 2.28590389, 0.76955451, 0.65880575,
3.08527118, 2.85095604, 0.90146894, 0.77070531])),
'1584.893192461114': [      fun: 0.9461414274572907
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
      jac: array([-0.04199981,  0.05929582, -0.06968102, -0.75312014,  2.16178828,
-1.08061848,  0.27109703, -0.00884198, -0.41823523])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
      nfev: 33
      nit: 11
      status: 0
success: True
      x: array([0.31170931, 0.95741517, 0.77139118, 1.70756672, 0.92246131,
0.75303595, 1.08758093, 1.01294321, 0.51097878])),
'1995.262314968881': [      fun: 0.9768351096886936
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
      jac: array([-0.05857852,  0.08655782, -0.09136404, -0.63239793,  1.87382128,
-1.26102655,  0.46400122, -0.01069632, -0.70464865])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
      nfev: 104
      nit: 12
      status: 0
success: True
      x: array([0.34179629, 0.94909154, 0.77097536, 1.7774106 , 0.93852622,
0.86730741, 1.26297248, 0.99681402, 0.41619756])),
'2511.886431509582': [      fun: 0.32064786907125614
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
      jac: array([ 7.72046843e-02, -1.09267041e-01, -1.31779149e-02, -1.57427901e-01,
2.98936112e-01,  4.93257611e-03, -2.80498028e-03, -1.99418653e-04,
-1.62456043e-02])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
      nfev: 123
      nit: 37
      status: 0
success: True
      x: array([0.99437932, 0.50004673, 3.12985663, 0.43298667, 0.38081053,
4.55725022, 5.73178498, 1.14314177, 0.62450296])),
'3162.2776601683795': [      fun: 3.944989752122985
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
      jac: array([ -0.8027215 ,  0.87737127, -1.22008855, 14.41677643,
-26.82816281, -7.38079209,  5.18666703,  0.2150572 ,
-3.38207205])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
      nfev: 21
      nit: 1
      status: 0
success: True
      x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,

```

```

        1.e-03]]],
'3981.0717055349733': [      fun: 0.2784001097419291
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-1.69091340e-03, -1.42178621e-04,  5.27232358e-05, -1.84827485e-05,
   -4.46309653e-05,  1.23039644e-04, -3.64949179e-05, -3.04526413e-04,
   -3.90352599e-04])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 71
    nit: 47
    status: 0
    success: True
      x: array([ 0.80299449,  9.33296634, 24.17669207,  5.31403233, 16.98522118,
  30.31623275,  1.10137553, 10.77297129,  2.05500571]))],
'5011.872336272725': [      fun: 4.351879145114877
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([ -0.87230637,  0.94613379, -1.15112769, 18.24990593,
   -34.77809417, -12.75341388, 10.77408397,  0.26239803,
   -4.73704303])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 72
    nit: 1
    status: 0
    success: True
      x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
  1.e-03]]],
'6309.573444801937': [      fun: 0.4212338374835501
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([ 0.14700358, -0.06610512, -0.00191661,  0.14965867, -0.26382057,
   -0.13747897, -0.17945971, -0.03066943, -0.1121963 ])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 62
    nit: 21
    status: 0
    success: True
      x: array([0.44581633, 1.08202368, 0.71299215, 1.89090932, 1.07343763,
  1.45149934, 0.11865993, 1.4586375 , 0.36982774]))],
'7943.282347242822': [      fun: 4.669747672319788
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([ -0.90573058,  0.98266857, -1.04147315, 22.0445503 ,
   -42.6942009 , -18.03786691, 16.32987646,  0.28441909,
   -6.12656177])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 49
    nit: 1
    status: 0
    success: True
      x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
  1.e-03]]],

```



```

'10000.0': [      fun: 3.622641106985052
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-0.21624069,  0.21070497, -0.31711513, -1.43965658,  4.37290847,
    5.90670681, -6.68048526,  0.02314986, -0.57371147])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 30
    nit: 9
    status: 0
    success: True
      x: array([0.30975999, 1.01781847, 0.73607879, 1.97560291, 1.01204801,
    1.00260385, 0.97117747, 1.05128972, 0.05881722]))],
'12589.254117941662': [      fun: 0.5811093554873149
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-0.03191147,  0.1040433 , -0.03655685, -7.69608823, 18.3032574 ,
    -5.94248578,  4.23649143, -0.04751259, -2.12870575])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 125
    nit: 41
    status: 0
    success: True
      x: array([0.90764324, 0.40760222, 1.13927688, 0.69816866, 0.40167333,
    0.54539913, 1.09033523, 1.19135967, 0.29569899]))],
'15848.93192461114': [      fun: 5.164061008479435
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-1.16500098,  1.27037642, -1.08280961, 34.49111123,
    -68.5995727 , -34.66415615, 33.55486117,  0.32400172,
    -10.35611186])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 68
    nit: 1
    status: 0
    success: True
      x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
    1.e-03]))],
'19952.62314968883': [      fun: 0.3569973459915211
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([ 1.41556579e-02,  1.01006973e-05,  3.96526391e-04, -5.23345286e-07,
    -7.41755660e-04,  1.32452371e-03,  1.53183768e-03, -4.59079856e-04,
    -8.60447487e-04])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 112
    nit: 20
    status: 0
    success: True
      x: array([ 1.12052173, 38.40697189, 62.416123 ,  0.18890671,
    33.03337692, 67.43839127,  1.02795281, 100. ,
    75.75180978]))],
'25118.864315095823': [      fun: 5.149578752631611

```

```

hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
  jac: array([ -0.84160183,  0.92470401, -0.71248669, 28.38305979,
 -56.07942491, -27.08765775, 26.04013081,  0.26712708,
 -8.72603034])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
  nfev: 23
  nit: 1
  status: 0
  success: True
    x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
 1.e-03]))],
'31622.776601683792': [      fun: 0.4190132412495196
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
  jac: array([-3.96162175e-06, -1.14625089e-05, 1.23532724e-05, 8.94347252e-07,
 1.45443805e-06, 1.13946245e-07, -2.11373636e-05, -1.09831757e-05,
 -3.46582598e-06])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
  nfev: 115
  nit: 38
  status: 0
  success: True
    x: array([ 1.26008098, 75.63134358, 54.1055075 ,  0.          ,
 10.20823561, 49.37465457,  0.26941967, 100.          ,
 86.51554565]))],
'39810.71705534969': [      fun: 5.636443073544105
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
  jac: array([ -0.73939077,  0.81277712, -0.596577 , 27.51890667,
 -51.35903121, -25.20893023, 24.00953362,  0.2248834 ,
 -8.10530069])
message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
  nfev: 62
  nit: 1
  status: 0
  success: True
    x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
 1.e-03]))],
'50118.72336272725': [      fun: 0.42925251751403254
hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
  jac: array([ 1.03184337e-06, -1.32306165e-04, 6.87847900e-06, -1.04829261e-06,
 -1.64640751e-06, -1.85202991e-08, 1.06651003e-02, -4.11321331e-04,
 1.65171117e-07])
message: b'CONVERGENCE: NORM_OF_PROJECTED_GRADIENT_<=_PGTOL'
  nfev: 123
  nit: 80
  status: 0
  success: True
    x: array([2.13835838e-02, 1.00000000e+02, 1.64350485e+00, 2.34444713e-04,
 9.47319045e-03, 8.60168354e+01, 0.00000000e+00, 1.00000000e+02,

```

```

7.57401563e+01]]],
'63095.73444801943': [      fun: 5.325453510213752
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-0.74029101,  0.81917288, -0.5141752 , 30.58766359,
-60.7943211 , -30.28964275, 29.5689715 ,  0.2271121 ,
-9.77175109])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 50
    nit: 1
    status: 0
    success: True
      x: array([3.e-01, 1.e+00, 7.e-01, 2.e+00, 1.e+00, 1.e+00, 1.e+00, 1.e+00,
1.e-03]))],
'79432.82347242821': [      fun: 4.287333239914811
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([-13.74513734,  5.92758425, -6.84171541, -2.36995273,
 9.35390639, -3.15010798,  6.67500811, -2.1420408 ,
-0.88699255])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 34
    nit: 3
    status: 0
    success: True
      x: array([0.36847101, 0.99123484, 0.73340712, 1.9921195 , 1.08465646,
1.07276962, 0.91895169, 1.04814585, 0.03444698]))],
'100000.0': [      fun: 0.5332888086776034
  hess_inv: <9x9 LbfgsInvHessProduct with dtype=float64>
    jac: array([ 1.74765831e+00, -4.70119020e-01,  1.82290997e-01,  4.16281354e-02,
 8.01897616e-03, -2.96084147e-01,  2.17136993e+00, -3.78069347e-03,
 8.04301435e-04])
  message: b'CONVERGENCE: REL_REDUCTION_OF_F_<=_FACTR*EPSMCH'
    nfev: 144
    nit: 77
    status: 0
    success: True
      x: array([ 0.          ,  0.74530929,  0.          ,  5.11425631,  0.74432465,
 0.54231939,  0.04349352, 11.16815885,  6.65153942]))]

```

## 0.1 Validation

In [10]: *## Validation of the outer jacobian*

```

ca = fitter.ca
dHdc = ca.hcat([ca.gradient(solver._inner_objective._obj_1, ci) for ci in model.cs]).re
d2Jdc2 = ca.hcat([ca.jacobian(solver._inner_objective.inner_jacobian, ci) for ci in mod
dJ2dcdp = ca.hcat([ca.jacobian(solver._inner_objective.inner_jacobian, pi) for pi in mo

```

```

In [11]: dHdc_fn = ca.Function("dhdccfn", solver._inner_objective.input_list, [dHdc])
d2Jdc2_fn = ca.Function("d2jdc2", solver._inner_objective.input_list, [d2Jdc2])

```

```

dJ2dcdp_fn = ca.Function("d2jdcdp", solver._inner_objective.input_list, [dJ2dcdp])

In [12]: in_arg = [model.observation_times, *fitter.argsplit(solver.problems[0].cache.recent, 3)]

In [13]: dhdc_eval = dHdc_fn(*in_arg)
          d2jdc2_eval = d2Jdc2_fn(*in_arg)
          dj2dcdp_eval = dJ2dcdp_fn(*in_arg)

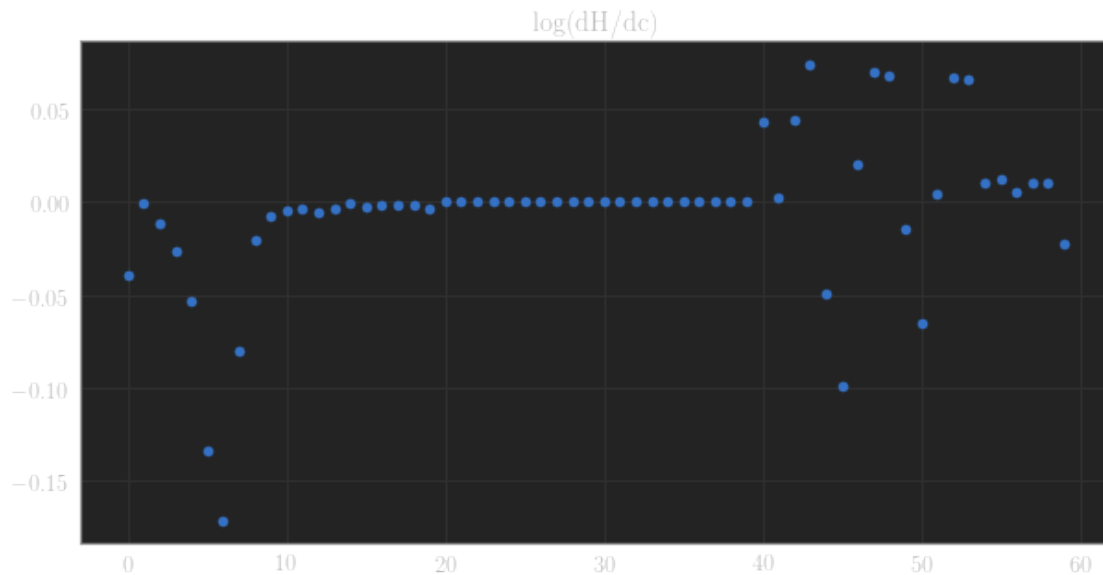
In [14]: import numpy as np
          def numerical_log(matrix):
              return np.log(np.fabs(np.array(matrix))+1e-16)

          import matplotlib.pyplot as plt
          plt.rcParams['figure.figsize'] = [10, 5]

In [15]: plt.plot(np.array(dhdc_eval).reshape(-1,), 'o')
          plt.title("log(dH/dc)")

Out[15]: Text(0.5, 1.0, 'log(dH/dc)')

```

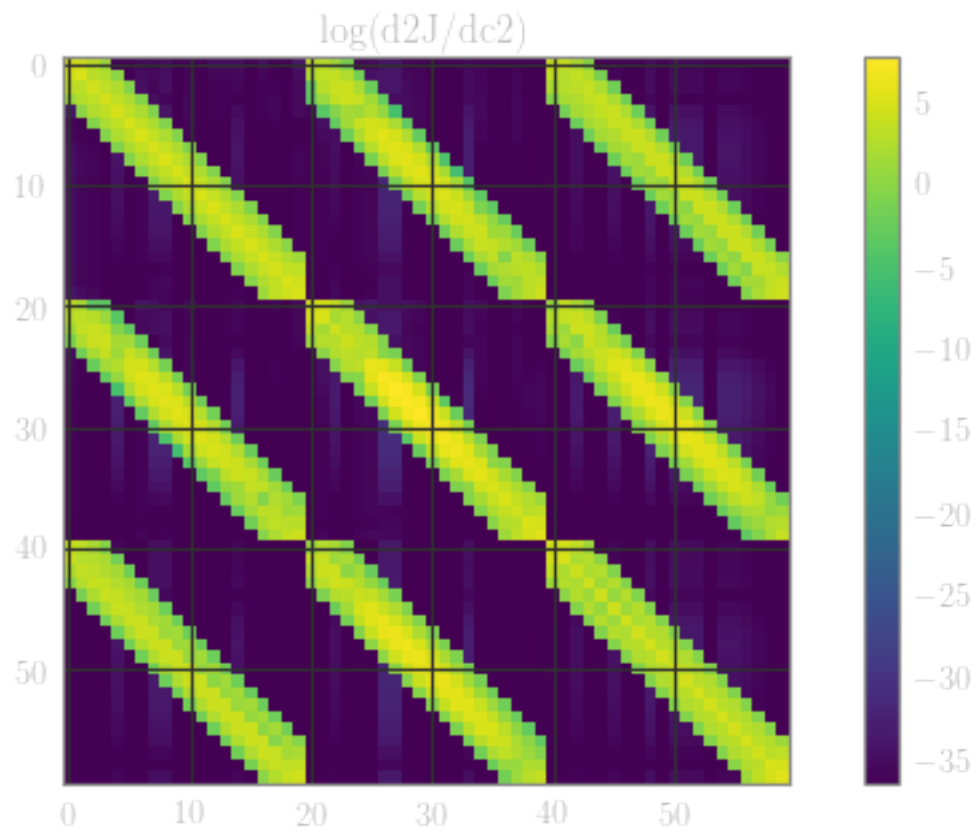


```

In [16]: plt.imshow(numerical_log(d2jdc2_eval))
          plt.colorbar()
          plt.title("log(d2J/dc2)")

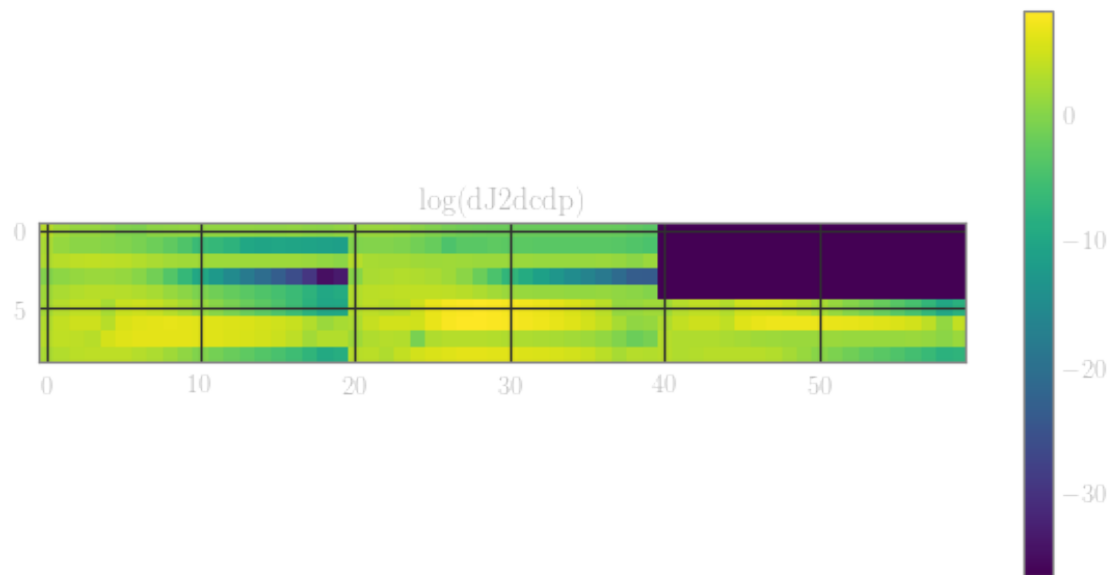
Out[16]: Text(0.5, 1.0, 'log(d2J/dc2)')

```



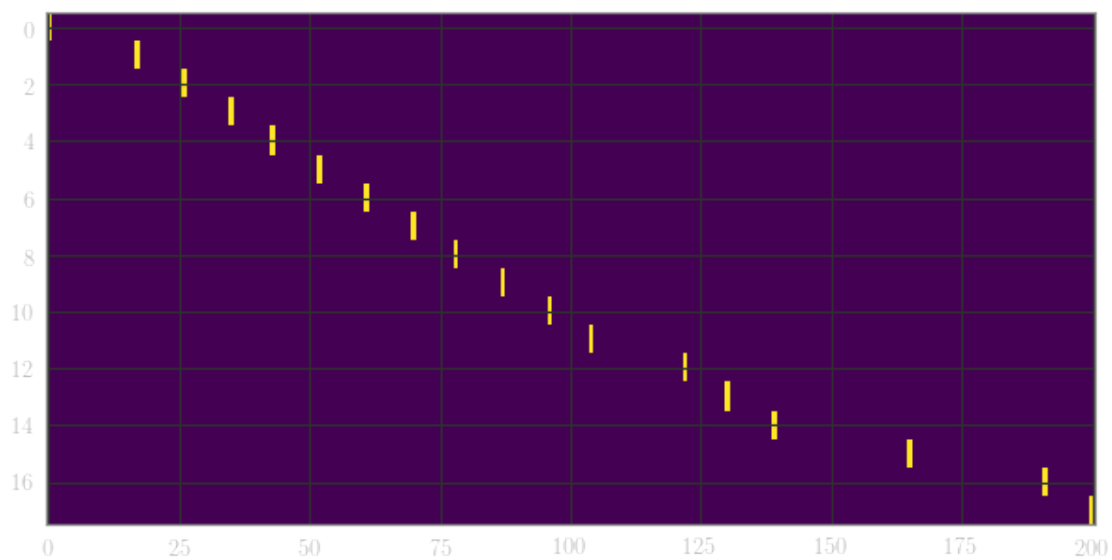
```
In [17]: plt.imshow(numerical_log(dj2dcdp_eval).T)
plt.colorbar()
plt.title("log(dJ2dcdp)")
```

```
Out[17]: Text(0.5, 1.0, 'log(dJ2dcdp)')
```



```
In [18]: H_num = solver._inner_objective.generate_collocation_matrix(context['datasets'][0], model)
plt.imshow(H_num, aspect='auto')
```

```
Out[18]: <matplotlib.image.AxesImage at 0x7f090b0fd278>
```



```
In [19]: # create and profile calls
```

```
obj_fn, obj_jac = solver._inner_objective.create_objective_functions(model, context['da
```

```
c_test = np.array(solver.problems[0].cache.recent)
```

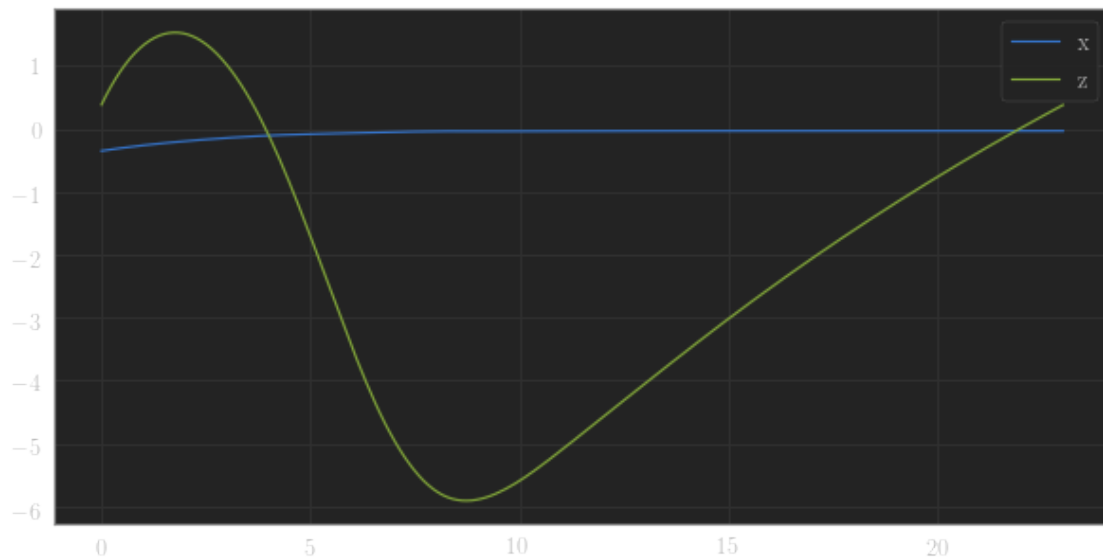
```
%timeit obj_fn(c_test, [0.3, 1, 0.7, 2, 1, 1, 1, 1, 1e-4], rho=1000)
```

34.3 ms ± 87.5 µs per loop (mean ± std. dev. of 7 runs, 10 loops each)

```
In [20]: getx = ca.Function("getx", [model.ts, *model.cs], model.xs)
```

```
In [21]: xs = np.array([np.array(i) for i in getx(model.observation_times, *fitter.argsplit(solv
plt.plot(model.observation_times, np.hstack([xs[0], xs[2]]))
plt.legend("xz")
```

```
iv = [xs[i][0].item() for i in range(3)]
```



```
In [22]: from scipy import integrate
```

```
sol = integrate.solve_ivp(lambda t, y: context['model'](t, y, [0.92255063, 0.55098101,
2.21360157, 2.85341497, 1.13601111, 0.55253756]), [0, 24], iv)
```

```
sol.y[2]
```

```
Out [22]: array([3.85342469e-01, 3.38503226e-01, 2.34002644e-01, 2.53160352e-01,
3.39436852e-01, 4.00043151e-01, 4.40658369e-01, 4.90852784e-01,
4.98479582e-01, 5.30750775e-01, 5.70088870e-01, 5.83474294e-01,
6.46924058e-01, 7.02786516e-01, 8.68126739e-01, 8.93082470e-01,
9.77807810e-01, 1.06132593e+00, 1.17267463e+00, 1.19303228e+00,
1.20622748e+00, 1.23523702e+00, 1.26081587e+00, 1.26457464e+00,
```

```

1.28595832e+00, 1.30243719e+00, 1.31543243e+00, 1.32466687e+00,
1.34275783e+00, 1.36403777e+00, 1.38544977e+00, 1.40985114e+00,
1.45740359e+00, 1.49191189e+00, 1.55587650e+00, 1.62297708e+00,
1.66173775e+00, 1.73268022e+00, 1.81420875e+00, 1.91495101e+00,
2.02242843e+00, 2.55109590e+00, 2.71421588e+00, 2.74464146e+00,
2.76589526e+00, 2.78715780e+00, 2.79942326e+00, 2.85068151e+00,
2.89182004e+00, 2.94044091e+00, 2.96578994e+00, 3.05170372e+00,
3.11682541e+00, 3.39394885e+00, 3.66410671e+00, 6.50540313e+00,
1.17096614e+01, 2.60474106e+01, 5.58184109e+01, 1.14063609e+02,
2.27268770e+02, 4.59551472e+02, 9.56163469e+02, 2.07767704e+03,
4.79367641e+03, 1.19591682e+04, 3.28723351e+04, 1.00895296e+05,
3.43779740e+05, 1.26688443e+06, 4.93890115e+06, 1.97371039e+07,
7.88720283e+07, 2.60576992e+08, 8.61202080e+08, 3.30034680e+09,
3.30413018e+09])

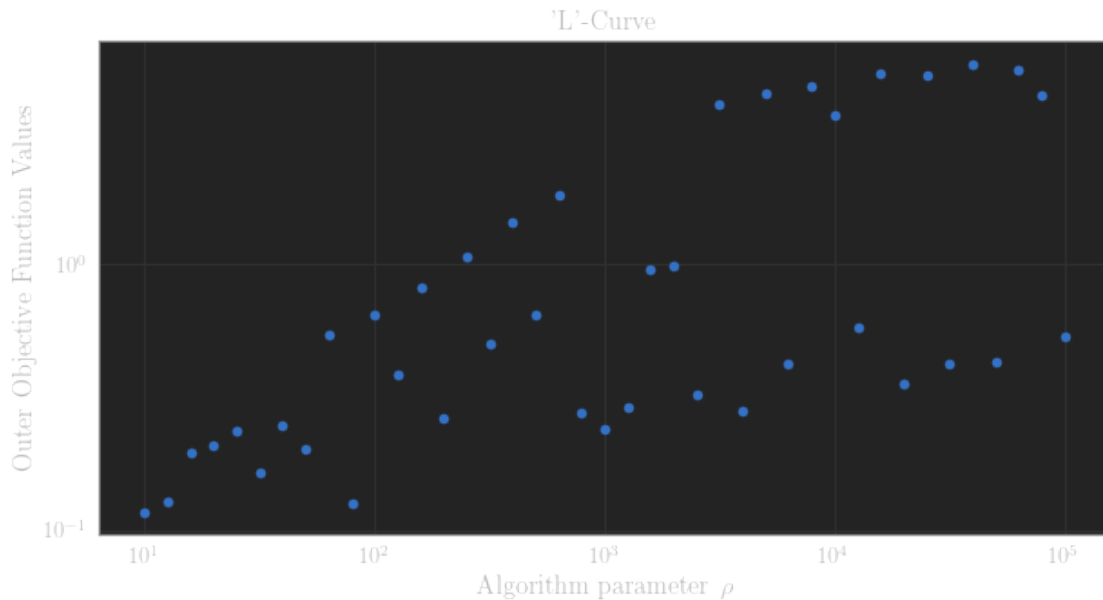
```

```
In [23]: outer_evals = {r:v[0].fun for r, v in solver.solutions.items()}
```

```
outer_list = np.array([[float(key), value] for key, value in outer_evals.items()])
```

```
In [24]: plt.loglog(*outer_list.T, 'o')
plt.xlabel(r"Algorithm parameter  $\rho$  ")
plt.ylabel(r"Outer Objective Function Values")
plt.title("'L'-Curve")
```

```
Out[24]: Text(0.5, 1.0, "'L'-Curve")
```



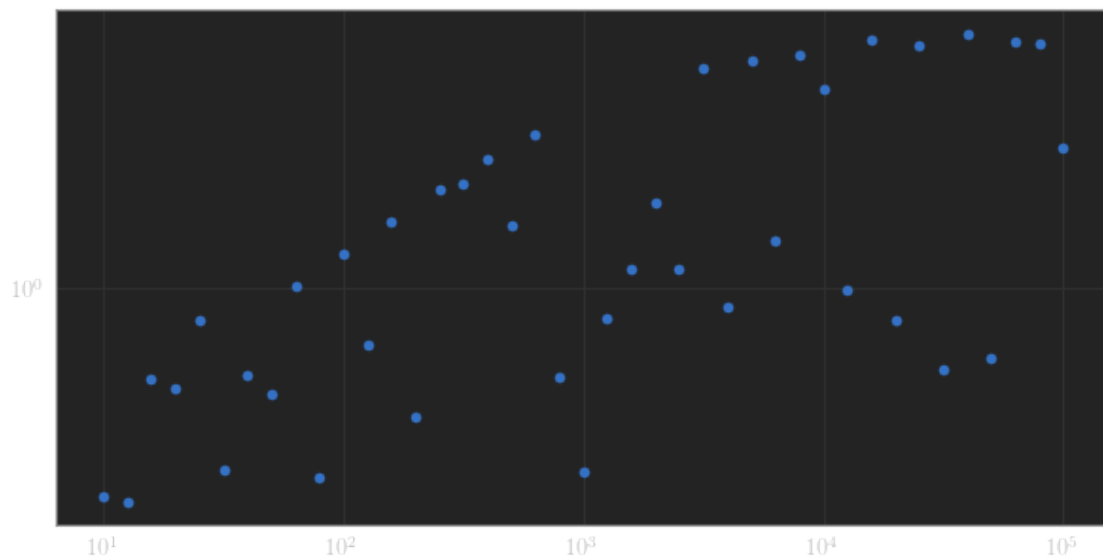
```
In [25]: def gen_key(sol_key, solution):
return "y".join(map(str,solution.x)) + "r" + sol_key
```



```
In [26]: inner_evals = dict()
        for soli in solver.solutions:
            key = gen_key(soli, solver.solutions[soli][0])
            inner_evals[soli] = solver.problems[0].cache.results[key]
```

```
In [27]: new_list = []
        for key in inner_evals.keys():
            new_list.append((float(key), inner_evals[key].fun))
        new_list = np.array(new_list)
        plt.loglog(*np.array(new_list).T, 'o')
```

```
Out[27]: [<matplotlib.lines.Line2D at 0x7f090af08b70>]
```



```
In [28]: diff_field_value = [[okey, (ivalue-ovalue)/(ikey)] for (ikey, ivalue), (okey, ovalue) i
```

```
In [29]: plt.loglog(*np.array(diff_field_value).T, 'o')
```

```
Out[29]: [<matplotlib.lines.Line2D at 0x7f090add9ef0>]
```

