test_functions_2

April 24, 2023

[ ]:

```
import torch
import numpy as np
import matplotlib.pyplot as plt
from torch import nn
from torch.autograd import Variable
import copy
```

[ ]:

```
d = 100

def synthetic_example(iters=100_000, lr=1e-3):
    # Objective function
    def func(x):
        val = 0
        for i in np.arange(d - 1):
            val += (100*(x[i + 1] - x[i]**2)**2 + (x[i] - 1)**2)
        return val

    x0 = np.random.uniform(-2.048, 2.048, d)

    x_Adam = Variable(torch.tensor(x0), requires_grad=True)
    x_AMS  = Variable(torch.tensor(x0), requires_grad=True)
    x_SGD  = Variable(torch.tensor(x0), requires_grad=True)

    # avg_regret_checkpoints = []
    # iteration_checkpoints  = []
    # x_checkpoints          = []

    optimizer_Adam = torch.optim.Adam([x_Adam], lr=lr, betas=(0.9, 0.999),␣
 ↪eps=1e-08, amsgrad=False)
    optimizer_AMSGrad = torch.optim.Adam([x_AMS], lr=lr, betas=(0.9, 0.999),␣
 ↪eps=1e-08, amsgrad=True)
    optimizer_SGD  = torch.optim.SGD([x_SGD], lr=lr, momentum=0.9, dampening=0,
                                     weight_decay=0, nesterov=True)

    # Create learning rate schedulers for Adam and AMSGrad
    # lambda1 = lambda iter: 1/np.sqrt(iter + 1)
    # scheduler_Adam = torch.optim.lr_scheduler.LambdaLR(optimizer_Adam,␣
↪lr_lambda=lambda1,
    #                                                    verbose=False)

    # scheduler_AMS  = torch.optim.lr_scheduler.LambdaLR(optimizer_AMSGrad,␣
↪lr_lambda=lambda1,
    #                                                    verbose=False)

    # lambda3 = lambda iter: 1/np.sqrt(iter + 1)
    # scheduler_SGD  = torch.optim.lr_scheduler.LambdaLR(optimizer_SGD,␣
↪lr_lambda=lambda3,
                                                    #    verbose=False)

    # total_regret = 0

    for iter in np.arange(1, iters + 1):
        loss_Adam     = func(x_Adam)
        loss_AMS      = func(x_AMS)
        loss_SGD      = func(x_SGD)

        # total_regret += np.linalg.norm(loss.item() - x_true)

        # if (iter % 10000 == 0):
        #     avg_regret = total_regret / iter
        #     avg_regret_checkpoints.append(avg_regret)
        #     iteration_checkpoints.append(iter)
        #     x_checkpoints.append(x.item())

        if (iter % 1000 == 0):
            print(f"Iteration: {iter}")
            print("---------------------------")
            print(f"f(x_Adam)  = {func(x_Adam)}")
            print(f"f(x_AMS)   = {func(x_AMS)}")
            print()

        optimizer_Adam.zero_grad()
        loss_Adam.backward()
        optimizer_Adam.step()
        # scheduler_Adam.step()

        optimizer_AMSGrad.zero_grad()
        loss_AMS.backward()
        optimizer_AMSGrad.step()
        # scheduler_AMS.step()

        # optimizer_SGD.zero_grad()
        # loss_SGD.backward()
        # optimizer_SGD.step()
        # scheduler_SGD.step()

    # return x_Adam, x_AMS, x_SGD
    return x_Adam, x_AMS
```

[ ]:

```
x_true = torch.tensor(np.ones(d))

iters = 100000
lr    = 1e-4
# x_Adam, x_AMS, x_SGD = synthetic_example(iters=iters, lr=lr)
x_Adam, x_AMS = synthetic_example(iters=iters, lr=lr)

print(f"2-norm between Adam x and true x:    {torch.linalg.vector_norm(x_Adam -␣
 ↪x_true)}")
print(f"2-norm between AMSGrad x and true x: {torch.linalg.vector_norm(x_AMS -␣
 ↪x_true)}")
# print(f"2-norm between SGD x and true x:     {torch.linalg.vector_norm(x_SGD␣
 ↪- x_true)}")
```

```
Iteration: 1000
---------------------------
f(x_Adam) = 38199.60279458601
f(x_AMS)  = 38199.65917359872

Iteration: 2000
---------------------------
f(x_Adam) = 30371.31290717461
f(x_AMS)  = 30391.078761130455

Iteration: 3000
---------------------------
f(x_Adam) = 24171.065660513737
f(x_AMS)  = 24461.802472582236

Iteration: 4000
---------------------------
f(x_Adam) = 19187.712687609197
f(x_AMS)  = 20057.223476134463

Iteration: 5000
---------------------------
f(x_Adam) = 15150.85474829237
f(x_AMS)  = 16740.69171423532

Iteration: 6000
---------------------------
f(x_Adam) = 11878.84576448723
```

```
f(x_AMS)  = 14190.1271067985

Iteration: 7000
---------------------------
f(x_Adam) = 9243.578125017459
f(x_AMS)  = 12190.223878544806

Iteration: 8000
---------------------------
f(x_Adam) = 7140.572218733155
f(x_AMS)  = 10595.811616944724

Iteration: 9000
---------------------------
f(x_Adam) = 5476.608969281301
f(x_AMS)  = 9306.50962257507

Iteration: 10000
---------------------------
f(x_Adam) = 4172.805037708087
f(x_AMS)  = 8251.006735351348

Iteration: 11000
---------------------------
f(x_Adam) = 3163.3603561937116
f(x_AMS)  = 7376.698642070656

Iteration: 12000
---------------------------
f(x_Adam) = 2391.547419183561
f(x_AMS)  = 6644.080720154493

Iteration: 13000
---------------------------
f(x_Adam) = 1806.0475898561986
f(x_AMS)  = 6023.432319018696

Iteration: 14000
---------------------------
f(x_Adam) = 1363.7497589436805
f(x_AMS)  = 5492.398574677154

Iteration: 15000
---------------------------
f(x_Adam) = 1031.826263565065
f(x_AMS)  = 5034.037194545653

Iteration: 16000
```

```
---------------------------
f(x_Adam) = 783.0599915065641
f(x_AMS)  = 4635.322006001947

Iteration: 17000
---------------------------
f(x_Adam) = 589.1044467590547
f(x_AMS)  = 4286.083229012028

Iteration: 18000
---------------------------
f(x_Adam) = 437.848770258129
f(x_AMS)  = 3978.283665217538

Iteration: 19000
---------------------------
f(x_Adam) = 325.62926044719256
f(x_AMS)  = 3705.5128004351154

Iteration: 20000
---------------------------
f(x_Adam) = 248.08369206540766
f(x_AMS)  = 3462.6070643145904

Iteration: 21000
---------------------------
f(x_Adam) = 195.01083671026515
f(x_AMS)  = 3245.3470884205426

Iteration: 22000
---------------------------
f(x_Adam) = 150.17476600645074
f(x_AMS)  = 3050.2264959650497

Iteration: 23000
---------------------------
f(x_Adam) = 118.89864999630196
f(x_AMS)  = 2874.2992832138443

Iteration: 24000
---------------------------
f(x_Adam) = 101.52394798026972
f(x_AMS)  = 2715.085333229016

Iteration: 25000
---------------------------
f(x_Adam) = 94.43221350444708
f(x_AMS)  = 2570.4986994808637
```

```
Iteration: 26000
---------------------------
f(x_Adam) = 92.4818772964133
f(x_AMS)  = 2438.7819405355795

Iteration: 27000
---------------------------
f(x_Adam) = 91.80095257931738
f(x_AMS)  = 2318.4470208495877

Iteration: 28000
---------------------------
f(x_Adam) = 91.20855229784107
f(x_AMS)  = 2208.225526037268

Iteration: 29000
---------------------------
f(x_Adam) = 90.50858029705927
f(x_AMS)  = 2107.0283695835656

Iteration: 30000
---------------------------
f(x_Adam) = 89.73256167376945
f(x_AMS)  = 2013.9136966315064

Iteration: 31000
---------------------------
f(x_Adam) = 88.93254137358662
f(x_AMS)  = 1928.0614212660937

Iteration: 32000
---------------------------
f(x_Adam) = 88.05000087937373
f(x_AMS)  = 1848.7530065719413

Iteration: 33000
---------------------------
f(x_Adam) = 87.33084431862306
f(x_AMS)  = 1775.3553532200328

Iteration: 34000
---------------------------
f(x_Adam) = 86.75431018278499
f(x_AMS)  = 1707.3078979904121

Iteration: 35000
---------------------------
```

```
f(x_Adam) = 86.14184310390192
f(x_AMS)  = 1644.1122184295803

Iteration: 36000
---------------------------
f(x_Adam) = 85.33045135913851
f(x_AMS)  = 1585.3235950372818

Iteration: 37000
---------------------------
f(x_Adam) = 84.29923456041314
f(x_AMS)  = 1530.5441035837498

Iteration: 38000
---------------------------
f(x_Adam) = 83.37029321195959
f(x_AMS)  = 1479.4169037950744

Iteration: 39000
---------------------------
f(x_Adam) = 82.33613367217619
f(x_AMS)  = 1431.6214630684394

Iteration: 40000
---------------------------
f(x_Adam) = 81.2184502073891
f(x_AMS)  = 1386.8695106863279

Iteration: 41000
---------------------------
f(x_Adam) = 80.11785265088615
f(x_AMS)  = 1344.9015634701857

Iteration: 42000
---------------------------
f(x_Adam) = 79.03884054464389
f(x_AMS)  = 1305.4839007729283

Iteration: 43000
---------------------------
f(x_Adam) = 77.81509134947062
f(x_AMS)  = 1268.4058967967858

Iteration: 44000
---------------------------
f(x_Adam) = 76.51754790468006
f(x_AMS)  = 1233.4776423702262
```

```
Iteration: 45000
---------------------------
f(x_Adam) = 75.37083499408308
f(x_AMS)  = 1200.5278071873477

Iteration: 46000
---------------------------
f(x_Adam) = 74.07821501654071
f(x_AMS)  = 1169.4017077663311

Iteration: 47000
---------------------------
f(x_Adam) = 72.83560971088443
f(x_AMS)  = 1139.9595567608762

Iteration: 48000
---------------------------
f(x_Adam) = 71.63602990945202
f(x_AMS)  = 1112.0748765315686

Iteration: 49000
---------------------------
f(x_Adam) = 70.36963247686415
f(x_AMS)  = 1085.633064759944

Iteration: 50000
---------------------------
f(x_Adam) = 69.14352913929243
f(x_AMS)  = 1060.5301029334034

Iteration: 51000
---------------------------
f(x_Adam) = 67.91104523147025
f(x_AMS)  = 1036.6714001519517

Iteration: 52000
---------------------------
f(x_Adam) = 66.66403278320243
f(x_AMS)  = 1013.9707651955423

Iteration: 53000
---------------------------
f(x_Adam) = 65.4348042907355
f(x_AMS)  = 992.3494993858459

Iteration: 54000
---------------------------
f(x_Adam) = 64.19488923961673
```

```
f(x_AMS)  = 971.7356017456367

Iteration: 55000
---------------------------
f(x_Adam) = 62.95638946230857
f(x_AMS)  = 952.0630766288557

Iteration: 56000
---------------------------
f(x_Adam) = 61.72245309251411
f(x_AMS)  = 933.271332730192

Iteration: 57000
---------------------------
f(x_Adam) = 60.48402830846546
f(x_AMS)  = 915.3046615259117

Iteration: 58000
---------------------------
f(x_Adam) = 59.24761710259384
f(x_AMS)  = 898.1117829883943

Iteration: 59000
---------------------------
f(x_Adam) = 58.010103869713724
f(x_AMS)  = 881.6454469411788

Iteration: 60000
---------------------------
f(x_Adam) = 56.7741304925409
f(x_AMS)  = 865.8620796025474

Iteration: 61000
---------------------------
f(x_Adam) = 55.53719638503864
f(x_AMS)  = 850.721466502719

Iteration: 62000
---------------------------
f(x_Adam) = 54.29960113630816
f(x_AMS)  = 836.1864647954596

Iteration: 63000
---------------------------
f(x_Adam) = 53.06236424435551
f(x_AMS)  = 822.2227397749054

Iteration: 64000
```

```
---------------------------
f(x_Adam) = 51.824823979571285
f(x_AMS)  = 808.7985219709292

Iteration: 65000
---------------------------
f(x_Adam) = 50.5880496038589
f(x_AMS)  = 795.8843824325405

Iteration: 66000
---------------------------
f(x_Adam) = 49.35148795677354
f(x_AMS)  = 783.4530246967294

Iteration: 67000
---------------------------
f(x_Adam) = 48.114040506186875
f(x_AMS)  = 771.4790925135843

Iteration: 68000
---------------------------
f(x_Adam) = 46.87683188899492
f(x_AMS)  = 759.9389927205723

Iteration: 69000
---------------------------
f(x_Adam) = 45.642410216294955
f(x_AMS)  = 748.8107328000449

Iteration: 70000
---------------------------
f(x_Adam) = 44.404632188851096
f(x_AMS)  = 738.0737726777812

Iteration: 71000
---------------------------
f(x_Adam) = 43.16614617984596
f(x_AMS)  = 727.7088902760167

Iteration: 72000
---------------------------
f(x_Adam) = 41.9304590682477
f(x_AMS)  = 717.6980602567777

Iteration: 73000
---------------------------
f(x_Adam) = 40.69271537322961
f(x_AMS)  = 708.0243453024817
```

```
Iteration: 74000
---------------------------
f(x_Adam) = 39.45566691305198
f(x_AMS)  = 698.6717991942995

Iteration: 75000
---------------------------
f(x_Adam) = 38.21931929191342
f(x_AMS)  = 689.6253808722098

Iteration: 76000
---------------------------
f(x_Adam) = 36.98120896142759
f(x_AMS)  = 680.8708785990998

Iteration: 77000
---------------------------
f(x_Adam) = 35.74452138170863
f(x_AMS)  = 672.3948433074853

Iteration: 78000
---------------------------
f(x_Adam) = 34.50787490550127
f(x_AMS)  = 664.1845301838274

Iteration: 79000
---------------------------
f(x_Adam) = 33.270827703518634
f(x_AMS)  = 656.2278475429696

Iteration: 80000
---------------------------
f(x_Adam) = 32.03291594607199
f(x_AMS)  = 648.5133120644003

Iteration: 81000
---------------------------
f(x_Adam) = 30.795641736589147
f(x_AMS)  = 641.0300095015762

Iteration: 82000
---------------------------
f(x_Adam) = 29.558344760055686
f(x_AMS)  = 633.7675600334784

Iteration: 83000
---------------------------
```

```
f(x_Adam) = 28.320429149132856
f(x_AMS)  = 626.7160875004889

Iteration: 84000
---------------------------
f(x_Adam) = 27.084144269902342
f(x_AMS)  = 619.8661918506433

Iteration: 85000
---------------------------
f(x_Adam) = 25.846787850660505
f(x_AMS)  = 613.208924212781

Iteration: 86000
---------------------------
f(x_Adam) = 24.608934151807944
f(x_AMS)  = 606.7357641057465

Iteration: 87000
---------------------------
f(x_Adam) = 23.372225582005086
f(x_AMS)  = 600.438598383397

Iteration: 88000
---------------------------
f(x_Adam) = 22.134893687426505
f(x_AMS)  = 594.3097016003886

Iteration: 89000
---------------------------
f(x_Adam) = 20.897553238336133
f(x_AMS)  = 588.3417175607846

Iteration: 90000
---------------------------
f(x_Adam) = 19.661484379311105
f(x_AMS)  = 582.5276418788225

Iteration: 91000
---------------------------
f(x_Adam) = 18.424598684426115
f(x_AMS)  = 576.8608054375898

Iteration: 92000
---------------------------
f(x_Adam) = 17.18802794756703
f(x_AMS)  = 571.3348586767943
```

```
Iteration: 93000
---------------------------
f(x_Adam) = 15.951172360641747
f(x_AMS)  = 565.9437566756051

Iteration: 94000
---------------------------
f(x_Adam) = 14.714763478656115
f(x_AMS)  = 560.6817450214799

Iteration: 95000
---------------------------
f(x_Adam) = 13.478191295574414
f(x_AMS)  = 555.54334647208

Iteration: 96000
---------------------------
f(x_Adam) = 12.24114382748599
f(x_AMS)  = 550.5233484260325

Iteration: 97000
---------------------------
f(x_Adam) = 11.003760257659899
f(x_AMS)  = 545.616791220631

Iteration: 98000
---------------------------
f(x_Adam) = 9.767686850283688
f(x_AMS)  = 540.8189572718112

Iteration: 99000
---------------------------
f(x_Adam) = 8.530822512336622
f(x_AMS)  = 536.1253610649288

Iteration: 100000
---------------------------
f(x_Adam) = 7.2936361607194105
f(x_AMS)  = 531.5317399950039

2-norm between Adam x and true x:    2.8573238587547274
2-norm between AMSGrad x and true x: 9.18164668240543
```

[ ]:

```
print(f"Infinity-norm between Adam x and true x:    {torch.linalg.
 ↪vector_norm(x_Adam - x_true, float('inf'))}")
print(f"Infinity-norm between AMSGrad x and true x: {torch.linalg.
 ↪vector_norm(x_AMS - x_true, float('inf'))}")
```

```
print()
# print(f"Infinity-norm between SGD x and true x:     {torch.linalg.
 ↪vector_norm(x_SGD - x_true, float('inf'))}")

print(x_Adam)
print()
print(x_AMS)
```

```
Infinity-norm between Adam x and true x:    0.9999047988576247
Infinity-norm between AMSGrad x and true x: 1.6766835948139

tensor([1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 9.9999e-01,
        1.0000e+00, 9.9999e-01, 1.0000e+00, 9.9999e-01, 1.0000e+00, 9.9999e-01,
        1.0000e+00, 9.9999e-01, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00, 1.0000e+00,
        1.0000e+00, 1.0000e+00, 9.9999e-01, 9.9998e-01, 9.9996e-01, 9.9993e-01,
        9.9986e-01, 9.9971e-01, 9.9942e-01, 9.9884e-01, 9.9768e-01, 9.9538e-01,
        9.9080e-01, 9.8180e-01, 9.6433e-01, 9.3118e-01, 8.7072e-01, 7.6700e-01,
        6.0323e-01, 3.7849e-01, 1.5490e-01, 3.4343e-02, 1.1295e-02, 1.0231e-02,
        1.0211e-02, 1.0199e-02, 1.0010e-02, 9.5201e-05], dtype=torch.float64,
       requires_grad=True)

tensor([-0.3451,  0.1344,  0.0298,  0.0145,  0.1129,  0.2597,  0.1252,  0.0531,
        -0.4478,  0.6925,  0.8171,  0.8009,  0.7074,  0.5324,  0.1935, -0.1842,
         0.0343, -0.2642,  0.0924,  0.0872,  0.0327,  0.0114,  0.0101,  0.0046,
        -0.6767,  0.7875,  0.7870,  0.6923,  0.5038,  0.2643,  0.0474, -0.3675,
         0.1802,  0.0209, -0.5324,  0.4447,  0.2601,  0.1044,  0.0183, -0.0549,
        -0.2911,  0.1357,  0.0300,  0.0111,  0.0143,  0.0062, -0.3034, -0.4554,
         0.6947,  0.8254,  0.8863,  0.8999,  0.8724,  0.7922,  0.6430,  0.3488,
         0.0348,  0.0068,  0.0102,  0.0130, -0.0707, -0.1039, -0.4127,  0.1552,
        -0.0765, -0.2940, -0.4134,  0.5176,  0.5367,  0.4258,  0.2711,  0.1546,
        -0.4622,  0.6533,  0.7524,  0.7510,  0.6656,  0.5103,  0.1595, -0.3497,
         0.0331,  0.0381,  0.1891,  0.0588,  0.0141,  0.0206, -0.1115,  0.0845,
         0.0262,  0.0346,  0.1845,  0.0706,  0.0333,  0.2615,  0.0511,  0.0305,
         0.0946, -0.1102,  0.1466, -0.0105], dtype=torch.float64,
       requires_grad=True)
```