ps2-1-solve

May 13, 2021

1 (a)

[1]:
```
import matplotlib.pyplot as plt
import numpy as np

import src.util as util
from src.p01_lr import logistic_regression
```

[2]:
```
ds1_a_path = 'data/ds1_a.csv'
ds1_b_path = 'data/ds1_b.csv'
```

[3]:
```
Xa, Ya = util.load_csv(ds1_a_path, add_intercept=True)
Xb, Yb = util.load_csv(ds1_b_path, add_intercept=True)
```

[4]:
```
logistic_regression(Xa, Ya)
```

```
Finished 10000 iterations
Finished 20000 iterations
Finished 30000 iterations
Converged in 30386 iterations
```

[5]:
```
logistic_regression(Xb, Yb)
```

```
Finished 10000 iterations
Finished 20000 iterations
Finished 30000 iterations
Finished 40000 iterations
Finished 50000 iterations
Finished 60000 iterations
Finished 70000 iterations
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-5-b602c8bbabd2> in <module>
----> 1 logistic_regression(Xb, Yb)

~/Course/Stanford.CS229.2018.autumn/Stanford.CS229.2018.autumn.ps/ps2/src/p01_lr.
↪py in logistic_regression(X, Y)
```

```
     30         if i % 10000 == 0:
     31             print('Finished %d iterations' % i)
---> 32         if np.linalg.norm(prev_theta - theta) < 1e-15:
     33             print('Converged in %d iterations' % i)
     34             break

<__array_function__ internals> in norm(*args, **kwargs)

~/anaconda3/envs/cs229/lib/python3.9/site-packages/numpy/linalg/linalg.py in␣
 ↪norm(x, ord, axis, keepdims)
   2524             (ord == 2 and ndim == 1)):
   2525
-> 2526             x = x.ravel(order='K')
   2527             if isComplexType(x.dtype.type):
   2528                 sqnorm = dot(x.real, x.real) + dot(x.imag, x.imag)

KeyboardInterrupt:
```

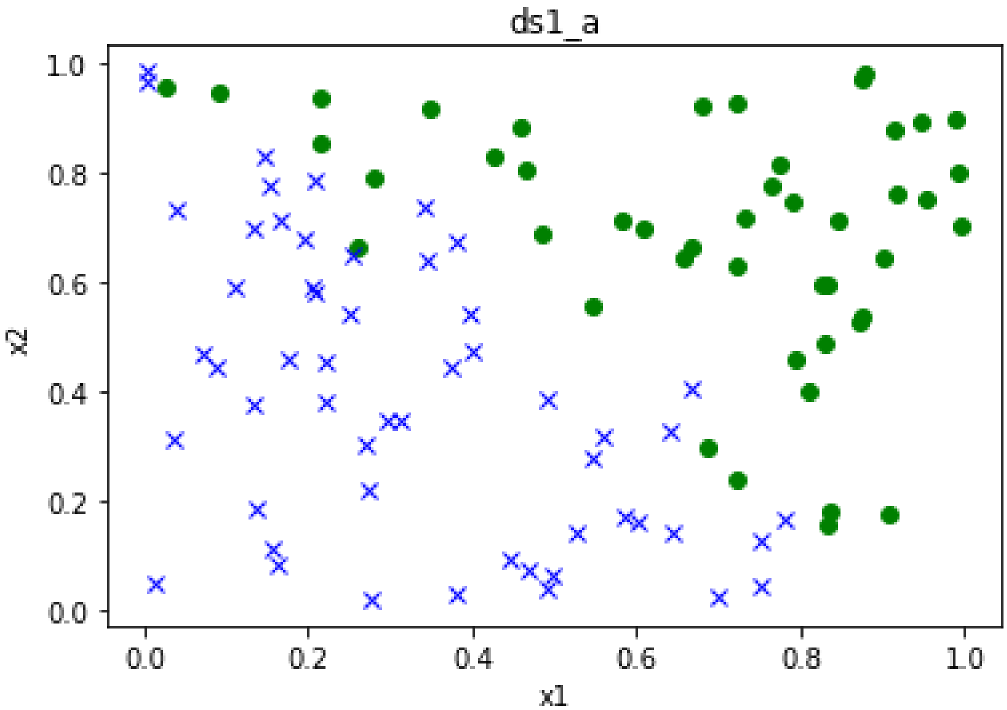
The logistic regression model spent too much time on data set B compared with A, it seems can't stop.

2 (b)

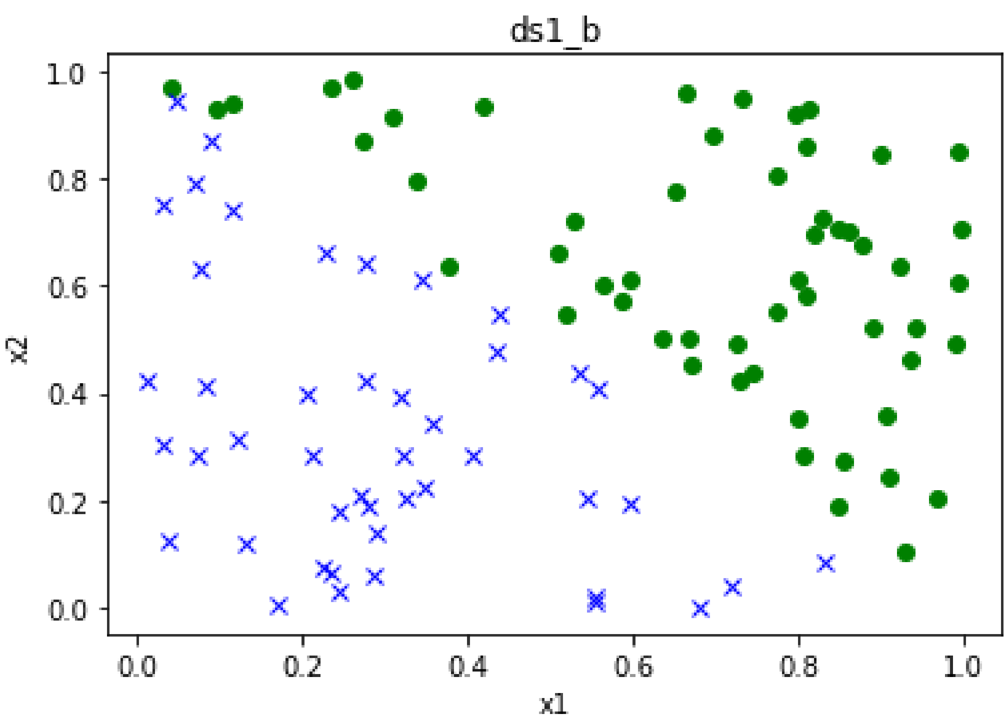
Check the data set.

[6]:
```python
def plot_data(x, y, title=None):
    plt.figure()
    plt.plot(x[y == 1, -2], x[y == 1, -1], 'go', linewidth=2)
    plt.plot(x[y == -1, -2], x[y == -1, -1], 'bx', linewidth=2)
    plt.xlabel('x1')
    plt.ylabel('x2')
    if not title == None: plt.title(title)
```

[7]:
```python
plot_data(Xa, Ya, 'ds1_a')
```



[8]:
```
plot_data(Xb, Yb, 'ds1_b')
```



Plot shows that ds_a and ds_b have similar distribution, so the problem is not on the data set.

Print parameters when algorithm learning.

[9]:
```
def logistic_regression_print(X, Y):
    '''
    A print version of logsitic regression
    which as same as algorithm in p01_lr.py
    but it can print some parameters when running
    '''

    def calc_grad(X, Y, theta):
        m, n = X.shape

        margins = Y * X.dot(theta)
        probs = 1. / (1 + np.exp(margins))
        grad = -(1./m) * (X.T.dot(probs * Y))

        return grad, probs, margins

    m, n = X.shape
    theta = np.zeros(n)
    learning_rate = 10

    i = 0
    while True:
        i += 1
        prev_theta = theta
        grad, probs, margins = calc_grad(X, Y, theta)
        theta = theta - learning_rate * grad
        theta_gap = np.linalg.norm(prev_theta - theta)
        if i % 10000 == 0:
            print('Finished %d iterations' % i)
            # print theta, gap, grad, probs, margins
            print('theta: ', theta, '\ntheta gap: ', theta_gap,
                  '\ngrad: ', grad, '\nprobs: ', probs,
                  '\nmargins: ', margins, '\n')
        if theta_gap < 1e-15:
            print('Converged in %d iterations' % i)
            break
    return
```

[10]:
```
logistic_regression_print(Xa, Ya)
```

```
Finished 10000 iterations
theta:  [-20.81394174  21.45250215  19.85155266]
```

```
theta gap:  7.226491864936692e-07
grad:  [ 4.15154546e-08 -4.27822247e-08 -4.08456455e-08]
probs:  [9.58459855e-03 2.35003503e-04 1.87805938e-02 3.38755801e-09
 3.01709647e-03 1.03898561e-01 5.60324684e-02 1.33905441e-04
 2.37284172e-01 2.13567063e-02 5.56258342e-06 5.30166197e-04
 9.06714952e-03 8.14262053e-04 2.22826669e-01 9.10762727e-03
 8.74194776e-05 7.90700913e-01 2.45242934e-08 4.74487689e-03
 1.24477629e-04 1.30627000e-03 1.53301676e-02 5.30848404e-07
 7.22499519e-02 7.96063684e-03 8.18675773e-03 6.64920586e-01
 6.01633832e-06 7.66266769e-05 1.25570829e-03 7.58067573e-01
 4.51099970e-01 7.35957110e-02 2.57845310e-05 4.94068037e-07
 7.64590847e-04 1.98089047e-06 6.58740596e-07 5.39220449e-01
 8.66873347e-07 4.66001810e-05 1.08453571e-05 1.70397093e-04
 8.34022182e-01 4.01492780e-02 5.15306053e-03 3.69852324e-02
 1.07050289e-01 7.89363846e-03 6.25534510e-06 2.82571508e-01
 1.81723860e-01 2.55095715e-07 2.69520265e-02 9.01082307e-08
 1.18942941e-01 3.41668362e-08 4.15261212e-05 1.43964556e-03
 1.07957516e-06 4.15935458e-02 9.91115665e-05 3.31475845e-01
 1.06423307e-02 1.97655239e-04 3.30494515e-01 1.66350026e-02
 3.26373154e-01 1.23324770e-03 2.26068811e-03 6.39002826e-01
 2.78300149e-05 1.45249028e-04 8.14428197e-02 7.71219167e-08
 1.31533834e-08 1.08527116e-04 4.95036114e-07 3.16219799e-01
 3.67816990e-01 4.15194145e-03 9.98578794e-02 3.22846762e-01
 3.82042131e-04 1.19246397e-05 6.01688791e-03 1.15911531e-03
 8.84017918e-01 1.57111583e-04 2.23636796e-03 1.72646436e-01
 5.16965480e-01 8.48916185e-02 1.66906579e-05 1.71784305e-05
 7.26746431e-04 7.63235858e-03 3.31251333e-08 1.66316832e-07]
margins:  [ 4.63796696  8.35567511  3.955972   19.50315652  5.80043869
2.15463857
  2.82416046  8.91824275  1.16762706  3.82480142 12.09944235  7.54178971
  4.69398883  7.11241372  1.2492692   4.6894937   9.34470502 -1.32915551
 17.52360162  5.34593362  8.99126006  6.63927241  4.16248376 14.44878882
  2.55263069  4.82525379  4.79701689 -0.68530005 12.02102572  9.47648865
  6.678799   -1.14211407  0.19622735  2.53272399 10.56571003 14.52059211
  7.17540482 13.1319621  14.23293535 -0.15720475 13.95837209  9.97385953
 11.43176264  8.67720859 -1.61440584  3.17417332  5.26299808  3.25955004
  2.12123155  4.83377314 11.98206799  0.93174164  1.50471158 15.18162675
  3.5863749  16.22225424  2.0024785  17.19201033 10.08914637  6.54191765
 13.73894189  3.13732695  9.21916529  0.70151759  4.5322164   8.52878859
  0.70594929  4.07947129  0.72463494  6.69687018  6.08982279 -0.57103875
 10.48936762  8.83691559  2.42290298 16.37787826 18.1465868   9.12840196
 14.51863462  0.77119898  0.54159342  5.48001865  2.19880469  0.74071981
  7.86959755 11.33689181  5.10715005  6.75893844 -2.03104162  8.75839716
  6.10066331  1.56698635 -0.06788798  2.37766714 11.00064471 10.97183883
  7.22620592  4.86769673 17.2229735  15.60937107]

Finished 20000 iterations
theta:  [-20.81437785  21.45295156  19.85198173]
```

```
theta gap:  5.3329818166826493e-11
grad:  [ 3.06369117e-12 -3.15719879e-12 -3.01426955e-12]
probs:  [9.58369542e-03 2.34960721e-04 1.87791726e-02 3.38617497e-09
 3.01670262e-03 1.03894145e-01 5.60296635e-02 1.33879358e-04
 2.37282082e-01 2.13550411e-02 5.56110480e-06 5.30084811e-04
 9.06633480e-03 8.14145527e-04 2.22824016e-01 9.10680375e-03
 8.74024561e-05 7.90703451e-01 2.45149677e-08 4.74431922e-03
 1.24454657e-04 1.30609079e-03 1.53288778e-02 5.30687712e-07
 7.22466396e-02 7.95989787e-03 8.18595900e-03 6.64925749e-01
 6.01482401e-06 7.66114897e-05 1.25552177e-03 7.58073736e-01
 4.51098454e-01 7.35924355e-02 2.57788920e-05 4.93912755e-07
 7.64469606e-04 1.98032091e-06 6.58545590e-07 5.39220299e-01
 8.66611008e-07 4.65907206e-05 1.08426566e-05 1.70364896e-04
 8.34027604e-01 4.01470558e-02 5.15249632e-03 3.69824770e-02
 1.07047010e-01 7.89275926e-03 6.25370662e-06 2.82565442e-01
 1.81720225e-01 2.55014889e-07 2.69501486e-02 9.00765470e-08
 1.18937777e-01 3.41540866e-08 4.15175791e-05 1.43943174e-03
 1.07926861e-06 4.15912958e-02 9.90925034e-05 3.31474442e-01
 1.06412751e-02 1.97620882e-04 3.30491720e-01 1.66337532e-02
 3.26367353e-01 1.23306688e-03 2.26038103e-03 6.39004872e-01
 2.78240311e-05 1.45222235e-04 8.14396578e-02 7.70946253e-08
 1.31482246e-08 1.08505328e-04 4.94880853e-07 3.16218531e-01
 3.67815338e-01 4.15150600e-03 9.98538790e-02 3.22842850e-01
 3.81981420e-04 1.19217046e-05 6.01618441e-03 1.15896017e-03
 8.84021390e-01 1.57081506e-04 2.23606205e-03 1.72643551e-01
 5.16962734e-01 8.48867903e-02 1.66866385e-05 1.71743107e-05
 7.26639682e-04 7.63149420e-03 3.31127842e-08 1.66262579e-07]
margins:  [ 4.6380621   8.35585721  3.95604912 19.50356488  5.80056963  2.154686
  2.82421349  8.91843759  1.16763861  3.82488109 12.0997082   7.54194332
  4.69407951  7.11255695  1.24928452  4.68958496  9.34489977 -1.32917085
 17.52398195  5.34605172  8.99144465  6.63940979  4.16256921 14.44909157
  2.55268011  4.82534736  4.79711526 -0.68532323 12.02127745  9.47668688
  6.67894773 -1.14214767  0.19623348  2.53277203 10.56592876 14.52090645
  7.17556353 13.13224967 14.23323143 -0.15720415 13.95867476  9.97406257
 11.43201167  8.6773976  -1.61444501  3.17423098  5.26310814  3.2596274
  2.12126585  4.83388542 11.98232996  0.93177156  1.50473602 15.18194365
  3.58644651 16.22260592  2.00252778 17.19238356 10.08935211  6.5420664
 13.73922588  3.1373834   9.21935767  0.70152392  4.53231666  8.52896246
  0.70596193  4.07954767  0.72466133  6.69701699  6.08995895 -0.57104762
 10.48958266  8.8371001   2.42294525 16.37823219 18.14697908  9.12860277
 14.51894831  0.77120485  0.54160052  5.48012397  2.1988492   0.7407377
  7.86975653 11.33713798  5.10726769  6.75907245 -2.03107548  8.75858864
  6.10080041  1.56700654 -0.06787698  2.37772929 11.00088556 10.97207868
  7.22635292  4.86781086 17.22334637 15.60969733]

Finished 30000 iterations
theta:  [-20.81437788  21.45295159  19.85198176]
theta gap:  6.153480596427404e-15
```

```
grad:  [ 1.89444920e-16 -2.86835475e-16 -1.92283988e-16]
probs:  [9.58369535e-03 2.34960718e-04 1.87791725e-02 3.38617487e-09
 3.01670259e-03 1.03894145e-01 5.60296633e-02 1.33879356e-04
 2.37282082e-01 2.13550410e-02 5.56110469e-06 5.30084805e-04
 9.06633474e-03 8.14145518e-04 2.22824015e-01 9.10680369e-03
 8.74024549e-05 7.90703451e-01 2.45149670e-08 4.74431918e-03
 1.24454655e-04 1.30609078e-03 1.53288777e-02 5.30687700e-07
 7.22466393e-02 7.95989782e-03 8.18595894e-03 6.64925749e-01
 6.01482389e-06 7.66114886e-05 1.25552176e-03 7.58073736e-01
 4.51098454e-01 7.35924353e-02 2.57788916e-05 4.93912744e-07
 7.64469597e-04 1.98032086e-06 6.58545576e-07 5.39220299e-01
 8.66610989e-07 4.65907199e-05 1.08426564e-05 1.70364893e-04
 8.34027604e-01 4.01470556e-02 5.15249628e-03 3.69824768e-02
 1.07047010e-01 7.89275920e-03 6.25370650e-06 2.82565441e-01
 1.81720225e-01 2.55014883e-07 2.69501484e-02 9.00765447e-08
 1.18937777e-01 3.41540857e-08 4.15175785e-05 1.43943173e-03
 1.07926859e-06 4.15912957e-02 9.90925020e-05 3.31474442e-01
 1.06412750e-02 1.97620879e-04 3.30491719e-01 1.66337531e-02
 3.26367352e-01 1.23306687e-03 2.26038100e-03 6.39004873e-01
 2.78240307e-05 1.45222233e-04 8.14396576e-02 7.70946233e-08
 1.31482243e-08 1.08505327e-04 4.94880842e-07 3.16218531e-01
 3.67815338e-01 4.15150597e-03 9.98538787e-02 3.22842850e-01
 3.81981416e-04 1.19217043e-05 6.01618435e-03 1.15896016e-03
 8.84021390e-01 1.57081504e-04 2.23606202e-03 1.72643551e-01
 5.16962733e-01 8.48867899e-02 1.66866382e-05 1.71743104e-05
 7.26639674e-04 7.63149414e-03 3.31127833e-08 1.66262575e-07]
margins:  [ 4.63806211  8.35585723  3.95604913 19.50356491  5.80056964  2.154686
  2.82421349  8.9184376   1.16763861  3.8248811  12.09970822  7.54194333
  4.69407952  7.11255696  1.24928453  4.68958496  9.34489978 -1.32917086
 17.52398198  5.34605173  8.99144466  6.6394098   4.16256922 14.44909159
  2.55268011  4.82534737  4.79711527 -0.68532323 12.02127747  9.4766869
  6.67894774 -1.14214767  0.19623348  2.53277203 10.56592878 14.52090647
  7.17556354 13.13224969 14.23323145 -0.15720415 13.95867478  9.97406259
 11.43201169  8.67739761 -1.61444502  3.17423099  5.26310815  3.25962741
  2.12126586  4.83388543 11.98232998  0.93177156  1.50473602 15.18194367
  3.58644651 16.22260594  2.00252778 17.19238359 10.08935212  6.54206641
 13.7392259   3.1373834   9.21935768  0.70152392  4.53231667  8.52896247
  0.70596193  4.07954768  0.72466134  6.697017    6.08995896 -0.57104762
 10.48958267  8.83710011  2.42294525 16.37823222 18.14697911  9.12860278
 14.51894833  0.77120485  0.54160052  5.48012398  2.1988492   0.7407377
  7.86975655 11.337138    5.10726769  6.75907246 -2.03107549  8.75858866
  6.10080042  1.56700655 -0.06787698  2.3777293  11.00088558 10.9720787
  7.22635293  4.86781087 17.22334639 15.60969735]

Converged in 30386 iterations
```

[11]:
```
logistic_regression_print(Xb, Yb)
```

```
Finished 10000 iterations
theta:  [-52.74109217  52.92982273  52.69691453]
theta gap:  0.003361039469518825
grad:  [ 0.00019399 -0.00019355 -0.00019461]
probs:  [1.60781163e-05 2.88793319e-01 7.22549611e-18 1.16553979e-10
 1.92932169e-15 1.51297562e-01 5.26859357e-08 4.06684209e-01
 9.11430490e-06 5.66397312e-14 1.32539864e-05 9.50880919e-10
 1.02231164e-09 6.77682921e-05 2.22506131e-06 1.24292660e-19
 2.31693797e-05 1.56851956e-04 6.48682666e-13 5.18759343e-16
 2.87901125e-03 1.49347534e-07 9.07050972e-04 4.23537260e-20
 1.09869887e-15 1.15678950e-13 4.08020388e-01 3.91003785e-16
 1.84623791e-10 4.99454209e-02 2.40851905e-16 1.27724850e-01
 2.53172373e-06 8.03274590e-09 3.16326808e-08 1.60086763e-13
 1.61249923e-06 1.14281111e-13 1.96244681e-07 1.36206493e-01
 3.33885561e-17 8.35632324e-17 2.01259813e-11 1.27845759e-04
 1.37578242e-14 1.24194868e-04 3.19818166e-06 1.71145315e-10
 1.22691608e-12 2.35946190e-07 1.80912737e-13 2.11759588e-01
 2.79698959e-12 1.31536204e-03 1.00820758e-03 6.76844895e-17
 6.80720008e-10 5.00811492e-04 1.51184652e-07 8.94926897e-12
 3.88464643e-10 8.81736813e-03 8.45784015e-18 8.20331006e-07
 1.01841007e-02 4.50413969e-04 1.01709098e-01 1.43618031e-02
 5.20037115e-14 2.85812558e-04 1.75583828e-13 6.80698077e-04
 3.19615021e-12 2.21359567e-04 6.52827397e-04 9.63538986e-17
 3.94457527e-10 2.90101156e-17 1.04825352e-01 1.19016928e-04
 1.46925799e-10 4.92237331e-14 7.37835315e-18 1.13545928e-05
 7.64283872e-08 1.69828524e-11 2.04731637e-13 3.39562161e-01
 4.60030739e-15 1.31901520e-13 2.76186643e-02 3.02303631e-04
 7.91334253e-06 8.48950798e-14 2.97325331e-04 1.74314833e-12
 5.72182260e-20 1.83959470e-01 9.62448846e-10 1.37957581e-02]
margins:  [11.03803537  0.90125181 39.46891578 22.87266661 33.88160791
1.72446013
 16.75891724  0.3776898  11.6056563  30.50206569 11.23119894 20.77363228
 20.70119946  9.59934837 13.01572386 43.531648   10.67265582  8.76005129
 28.06383276 35.19509159  5.8474252  15.71698966  7.00440445 44.60823055
 34.44464976 29.78795771  0.37215505 35.47781443 22.41270092  2.94558861
 35.96234943  1.92122657 12.88660764 18.6397394  17.26907502 29.46306046
 13.33772366 29.80011509 15.44390339  1.84716168 37.93831846 37.02092805
 24.62900953  8.96455818 31.9171687   8.99353451 12.65292494 22.48850813
 27.42651734 15.25966183 29.34076159  1.31435153 26.60247743  6.63232711
  6.89857248 37.23167463 21.10787004  7.59877985 15.70476374 25.43944927
 21.66881896  4.72217538 39.31143783 14.01355709  4.57669121  7.70489295
  2.1783772   4.22871723 30.5874613   8.1598885  29.37065982  7.29171077
 26.46907409  8.41550079  7.33354475 36.87850382 21.65350964 38.07888709
  2.14472319  9.0361258  22.64109342 30.64240051 39.44798121 11.38587689
 16.38691157 24.79881696 29.21707636  0.66524598 33.01265327 29.65672081
  3.56125626  8.10377629 11.74695238 30.09736026  8.12038626 27.07532825
 44.30741447  1.48974856 20.7615402   4.26950232]
```

```
Finished 20000 iterations
theta:  [-68.10040977  68.26496086  68.09888223]
theta gap:  0.002173205351041188
grad:  [ 0.00012541 -0.00012529 -0.0001257 ]
probs:  [6.22679572e-07 2.37202790e-01 7.53598516e-23 1.42757214e-13
 1.00950443e-19 9.55250400e-02 3.78944649e-10 3.92009175e-01
 3.19699536e-07 7.77852786e-18 5.09911237e-07 2.25451256e-12
 2.43757015e-12 4.28522052e-06 4.86707464e-08 3.82450758e-25
 9.99244055e-07 1.23689185e-05 1.80808308e-16 1.85995635e-20
 5.35488186e-04 1.53780034e-09 1.16004213e-04 9.96020176e-26
 4.73922645e-20 1.99027016e-17 3.70259997e-01 1.29344547e-20
 2.58709145e-13 2.09089000e-02 6.84237748e-21 8.13523196e-02
 5.73713134e-08 3.46052897e-11 2.13322053e-10 3.11904504e-17
 3.32081686e-08 1.98776643e-17 2.24845620e-09 8.96848234e-02
 5.37110814e-22 1.71180892e-21 1.61561492e-14 1.00268848e-05
 1.32123775e-18 9.27601441e-06 7.59939936e-08 2.41153605e-13
 4.12525731e-16 2.76173303e-09 3.60744821e-17 1.52475728e-01
 1.22707629e-15 1.96214727e-04 1.42603382e-04 1.37509034e-21
 1.52784598e-12 5.33611290e-05 1.54578825e-09 5.70371258e-15
 7.26608266e-13 2.35331342e-03 9.12912343e-23 1.45883624e-08
 2.68495784e-03 4.93324938e-05 5.69964488e-02 4.29187511e-03
 7.07663134e-18 2.77350910e-05 3.49393262e-17 8.32996990e-05
 1.43305912e-15 1.93699725e-05 8.01937215e-05 2.05451591e-21
 7.50084929e-13 4.34967032e-22 6.13244559e-02 8.59527355e-06
 2.02781831e-13 6.65936505e-18 7.52746833e-23 4.28582213e-07
 6.35966550e-10 1.22174412e-14 4.24406717e-17 2.96400562e-01
 3.06804283e-19 2.40839367e-17 1.00720354e-02 2.95362945e-05
 2.51868697e-07 1.30433574e-17 2.96190749e-05 6.73057917e-16
 1.42871114e-25 1.22640491e-01 2.34971658e-12 3.78142433e-03]
margins:  [14.28923316  1.16807679 50.93976757 29.57763101 43.73965722
2.24796621
 21.69363097  0.43887454 14.95588391 39.39516457 14.48902866 26.81808733
 26.74001941 12.36033426 16.83818763 56.2231976  13.81626579 11.30031144
 36.24909428 45.43114884  7.5317961  20.29291279  9.06176804 57.56861509
 44.49582793 38.45567619  0.53110159 45.7943923  28.98307196  3.84644979
 46.4311517   2.42411333 16.67372137 24.08701457 22.26821811 38.0064197
 17.22046992 38.45693497 19.91302199  2.31748933 48.9758378  47.81673629
 31.75647567 11.51023057 41.16796268 11.58806931 16.39261146 29.0533423
 35.42423309 19.70740744 37.86094593  1.71531406 34.33414205  8.53610472
  8.85530072 48.03576752 27.20716223  9.83837464 20.28773186 32.7976591
 27.9503889   6.04957489 50.74798746 18.04304171  5.91740168  9.91687826
  2.80608108  5.44673044 39.48973368 10.49278439 37.89291865  9.39298232
 34.17896499 10.85176713  9.43098513 47.6342467  27.91858996 49.18677199
  2.72829117 11.6642895  29.22664572 39.55050753 50.94089836 14.66278283
 21.17587515 32.03591186 37.69842453  0.86449743 42.62807692 38.26498658
  4.58786936 10.42986119 15.19435768 38.87825268 10.42706236 34.93470029
 57.20785459  1.9676596  26.7767264   5.57386594]
```

```
Finished 30000 iterations
theta:  [-79.01759142  79.17745526  79.03755803]
theta gap:  0.0016644896054955345
grad:  [ 9.60445104e-05 -9.60553587e-05 -9.61981110e-05]
probs:  [6.21102878e-08 2.04716545e-01 2.16081688e-26 1.22420835e-15
 9.12851810e-23 6.80698496e-02 1.14549278e-11 3.79864732e-01
 2.94131349e-08 1.40082136e-20 5.02297771e-08 3.06773133e-14
 3.33829158e-14 5.98860206e-07 3.23222204e-09 4.63224409e-29
 1.07579047e-07 2.02979393e-06 5.38714731e-19 1.28468860e-23
 1.61316078e-04 5.94654613e-11 2.69553559e-05 9.89581823e-30
 3.75307906e-23 4.19538963e-20 3.46142089e-01 8.43508945e-24
 2.44102647e-15 1.12072812e-02 4.01164961e-24 5.78402566e-02
 3.90860730e-09 7.22678590e-13 6.07862189e-12 7.14911412e-20
 2.10181123e-09 4.21305414e-20 9.33991977e-11 6.52249794e-02
 2.09971488e-25 7.97789801e-25 1.01176983e-16 1.62514311e-06
 1.82951144e-21 1.46144700e-06 5.37202507e-09 2.27438965e-15
 1.40394826e-18 1.17073286e-10 8.42248053e-20 1.19469656e-01
 5.02465163e-18 5.04926131e-05 3.52025044e-05 6.31805758e-25
 1.98619221e-14 1.09081593e-05 5.95757311e-11 3.02907993e-17
 8.32291803e-15 9.10500078e-04 2.68714147e-26 8.25306403e-10
 1.03829106e-03 1.01889722e-05 3.72073217e-02 1.80634874e-03
 1.26307159e-20 5.24717689e-06 8.13688156e-20 1.86430137e-05
 5.97577191e-18 3.42059964e-06 1.79500809e-05 9.85388576e-25
 8.67658056e-15 1.62490581e-25 4.11067328e-02 1.32666032e-06
 1.87759041e-15 1.18177329e-20 2.13414994e-26 4.14762399e-08
 2.11967709e-11 7.14486016e-17 1.01816183e-19 2.67916870e-01
 3.30212781e-22 5.27845912e-20 4.87280587e-03 5.62354955e-06
 2.18186785e-08 2.54624599e-20 5.69573923e-06 2.51306988e-18
 1.48767162e-29 9.10945531e-02 3.25141012e-14 1.51418837e-03]
margins:  [16.59435413  1.35707228 59.09672608 34.33648201 50.74805377
2.61672349
 25.1926011   0.49012241 17.34182447 45.71464311 16.80665777 31.11525299
 31.03073213 14.32823705 19.550096   65.24192626 16.04503983 13.10757425
 42.06510078 52.70894079  8.73198357 23.54562545 10.52130159 66.78544052
 51.63688055 44.61771564  0.63604071 53.12964191 33.64635776  4.47992105
 53.8728397   2.79048983 19.36008471 27.95581182 25.82624311 44.08471341
 19.98046637 44.61351403 23.09413836  2.66246337 56.82282576 55.48795236
 36.82966038 13.32991305 47.750238   13.43608206 19.04206089 33.71706466
 41.10724322 22.868221   43.92079748  1.99746198 39.83217555  9.89363301
 10.25435813 55.72121551 31.54997196 11.42598858 23.54377282 38.03568766
 32.41976348  7.00060566 58.87873444 20.9152664   6.8691403  11.49419439
  3.25333254  6.31463976 45.81815534 12.15781511 43.95529485 10.89002044
 39.65881839 12.58569127 10.92789798 55.27676145 32.37814889 57.07917747
  3.14960785 13.53284448 33.90878714 45.88468576 59.10914401 16.99814506
 24.57717226 37.17755334 43.73111789  1.00521733 49.46230499 44.38806764
  5.31920064 12.08854188 17.6404994  45.11708174 12.07578647 40.52502661
 66.37775547  2.30034306 31.05710252  6.49136038]
```

```
Finished 40000 iterations
theta:  [-87.70771189  87.87276307  87.73897393]
theta gap:  0.00137040952812708
grad:  [ 7.90651647e-05 -7.91523106e-05 -7.91443884e-05]
probs:  [9.95653955e-09 1.81416108e-01 3.26157351e-29 2.78521339e-17
 3.44500168e-25 5.17753960e-02 7.11275605e-13 3.69119067e-01
 4.38703772e-09 9.16757456e-23 7.92499006e-09 1.00251810e-15
 1.09841026e-15 1.24520584e-07 3.74620811e-10 3.53740278e-32
 1.83140851e-08 4.80871900e-07 5.26493971e-21 3.91056328e-26
 6.19487469e-05 4.46492745e-12 8.44876574e-06 6.41964913e-33
 1.27850201e-25 3.10631294e-22 3.28648770e-01 2.45106979e-26
 5.99215865e-17 6.82892209e-03 1.07230929e-26 4.35802921e-02
 4.62195387e-10 3.32843352e-14 3.56624607e-13 5.63687694e-22
 2.33476415e-10 3.12803491e-22 7.40199029e-12 4.99861475e-02
 4.06181517e-28 1.78094197e-27 1.77403446e-18 3.78968020e-07
 9.65036248e-24 3.34672936e-07 6.56007303e-10 5.56432748e-17
 1.52500691e-20 9.46272104e-12 6.74828149e-22 9.79854899e-02
 6.30574873e-20 1.70803983e-05 1.14898437e-05 1.38572404e-27
 6.22689607e-16 3.09065163e-06 4.46655086e-12 4.65534388e-19
 2.36412093e-16 4.24935610e-04 4.14677658e-29 8.33846660e-11
 4.87547462e-04 2.89356977e-06 2.63436260e-02 9.04936289e-04
 8.18911080e-23 1.38716277e-06 6.50164443e-22 5.64685690e-06
 7.62291724e-20 8.58738032e-07 5.43081020e-06 2.25091093e-27
 2.48064376e-16 3.04305848e-28 2.96205440e-02 2.99573944e-07
 4.51261836e-17 7.61850637e-23 3.20402234e-29 6.43721020e-09
 1.41678635e-12 1.19479221e-18 8.33345534e-22 2.46588427e-01
 1.43240468e-24 4.01989022e-22 2.72441403e-03 1.49575508e-06
 3.12110532e-09 1.77796099e-22 1.52296573e-06 2.92568814e-20
 1.00537654e-32 7.17169937e-02 1.07325333e-15 7.35088191e-04]
margins:  [18.42503625  1.50678256 65.59275795 38.11962209 56.32770293
2.90767634
 27.97171641  0.53599788 19.24461161 50.74378438 18.65324476 34.53626146
 34.44491248 15.89879468 21.70510677 72.4193302  17.81559538 14.54766444
 46.69321726 58.50353099  9.68914123 26.13476815 11.68148174 74.1259446
 57.31893824 49.52343557  0.71430283 58.97068784 37.35349486  4.97973609
 59.79739788  3.08859181 21.4950334  31.03368952 28.66209269 48.92754187
 22.17794005 49.51646706 25.62927219  2.94473065 63.07075264 61.59265509
 40.87327537 14.78581364 52.99504675 14.91011176 21.14484919 37.42757045
 45.62970292 25.38366114 48.74758417  2.2198112  44.21024015 10.97756197
 11.37403558 61.84357474 35.0124835  12.68712551 26.13440463 42.21110099
 35.98095523  7.76314788 65.35263639 23.20755668  7.62563525 12.75301671
  3.60983209  7.00674067 50.85665182 13.48824868 48.78481691 12.08440582
 44.02054272 13.96780107 12.1234168  61.35846252 35.93284338 63.35951952
  3.48921901 15.02090426 37.63706903 50.9288768  65.6105607  18.86117058
 27.28262995 41.26855939 48.53659387  1.11689099 54.90268761 49.26561745
  5.90277378 13.41287791 19.58507862 50.08140485 13.39484946 44.97817214
 73.67736084  2.56060892 34.46808186  7.21478472]
```

```
Finished 50000 iterations
theta:  [-95.01838735  95.1948202   95.0551918 ]
theta gap:  0.0011758957242939641
grad:  [ 6.78328688e-05 -6.79700947e-05 -6.78680755e-05]
probs:  [2.14162928e-09 1.63581405e-01 1.37653127e-31 1.15970585e-18
 3.15051733e-27 4.10848462e-02 6.89837376e-14 3.59391022e-01
 8.82314840e-10 1.33437219e-24 1.67330594e-09 5.63841623e-17
 6.22116509e-17 3.31062703e-08 6.13135469e-11 8.45081730e-35
 4.13864177e-09 1.42958092e-07 1.07404949e-22 2.98356437e-28
 2.76632704e-05 5.05797686e-13 3.18607054e-06 1.33063078e-35
 1.07524502e-27 5.00896168e-24 3.15055207e-01 1.79606586e-28
 2.65962561e-18 4.50835380e-03 7.33104566e-29 3.40732150e-02
 7.68615896e-11 2.50062369e-15 3.27129863e-14 9.54941683e-24
 3.67295369e-11 5.04979633e-24 8.75546860e-13 3.96088178e-02
 2.11433076e-30 1.04912432e-29 5.88349958e-20 1.10684790e-07
 1.16505516e-25 9.65857444e-08 1.12399568e-10 2.45720472e-18
 3.39969128e-22 1.14100017e-12 1.15993441e-23 8.28048948e-02
 1.58394968e-21 6.84261590e-06 4.45763363e-06 7.99893004e-30
 3.36648088e-17 1.07138581e-06 5.05785183e-13 1.38127850e-20
 1.17811444e-17 2.22766411e-04 1.78469820e-31 1.20625956e-11
 2.58361081e-04 1.00137045e-06 1.96516271e-02 5.05446012e-04
 1.17976188e-24 4.51045698e-07 1.11436936e-23 2.06234932e-06
 1.94402141e-21 2.67976132e-07 1.98120685e-06 1.35204143e-29
 1.24173877e-17 1.54686300e-30 2.23631210e-02 8.55897079e-08
 1.95718293e-18 1.09140726e-24 1.35074981e-31 1.33918128e-09
 1.45747827e-13 3.83064129e-20 1.45798249e-23 2.29530976e-01
 1.47239001e-26 6.61861294e-24 1.66680890e-03 4.89236687e-07
 6.08799194e-10 2.73418967e-24 4.99363661e-07 6.88579727e-22
 2.16415886e-35 5.86044452e-02 6.06777042e-17 4.02308076e-04]
margins:  [19.96169895  1.63181844 71.06057112 41.29836528 61.02223084
3.15016325
 30.30490561  0.57800826 20.84847216 54.97358132 20.20846456 37.41434336
 37.31598938 17.2235431  23.5150203  78.4562151  19.30289817 15.76071417
 50.58543597 63.37926392 10.49537734 28.31263964 12.65671902 80.30482516
 62.09724895 53.65081359  0.77659036 63.88678397 40.46834631  5.39730466
 64.78284954  3.34457645 23.28901485 33.62223622 31.05100426 53.00556214
 24.02743986 53.64269432 27.76392772  3.18828892 68.32881445 66.72701186
 44.27955011 16.01657929 57.41185889 16.15283458 22.90896102 40.54750726
 49.43318742 27.49911589 52.81109368  2.40483304 47.89436543 11.89233362
 12.32088805 66.998245   37.93007863 13.74655653 28.31266435 45.72869234
 38.98003135  8.40916403 70.80088856 25.14091173  8.26089402 13.81414005
  3.90974785  7.58956375 55.09672961 14.61169673 52.85116849 13.09166272
 47.68952824 15.13236765 13.1318024  66.47335208 38.92743395 68.64132378
  3.77772509 16.27370071 40.77502552 55.1745743  71.07947803 20.4312074
 29.55689848 44.70866963 52.58240351  1.21096146 59.48032548 53.37215641
  6.39517612 14.53041895 21.21953263 54.25620712 14.50993073 48.72741112
 79.81844649  2.77655286 37.34095536  7.81789002]
```

```
Finished 60000 iterations
theta:  [-101.37921493  101.57119731  101.41805781]
theta gap:  0.0010366393703263115
grad:  [ 5.97905761e-05 -5.99622968e-05 -5.97981756e-05]
probs:  [5.64049386e-10 1.49356417e-01 1.17894722e-33 7.32073893e-20
 5.30047420e-29 3.35969764e-02 9.09446081e-15 3.50480235e-01
 2.17971402e-10 3.36791288e-26 4.31703894e-10 4.60945629e-18
 5.12190892e-18 1.04239272e-08 1.27273226e-11 4.42759969e-37
 1.13619293e-09 4.96805576e-08 3.63700504e-24 4.28613683e-30
 1.37094192e-05 7.60674071e-14 1.36419971e-06 6.13550136e-38
 1.68488561e-29 1.38058611e-25 3.04001036e-01 2.48742252e-30
 1.77482547e-19 3.14541331e-03 9.56500338e-31 2.73432004e-02
 1.61574159e-11 2.63038430e-16 4.08135657e-15 2.73865144e-25
 7.34049226e-12 1.39197678e-25 1.36487065e-13 3.21348797e-02
 2.17513565e-32 1.20554185e-31 3.02577295e-21 3.77484024e-08
 2.48699766e-27 3.26816427e-08 2.43151121e-11 1.62968603e-19
 1.24348339e-23 1.81230660e-13 3.37143956e-25 7.14912547e-02
 6.41712405e-23 3.07940410e-06 1.94785305e-06 8.98808076e-32
 2.64838132e-18 4.26577977e-07 7.60903324e-14 6.44570619e-22
 8.64397351e-19 1.26518515e-04 1.55589279e-33 2.23408171e-12
 1.48838928e-04 3.97207299e-07 1.52103035e-02 3.04396330e-04
 2.94419137e-26 1.69111885e-07 3.22965952e-25 8.56648993e-07
 7.98887920e-23 9.71173789e-08 8.22448620e-07 1.58030394e-31
 9.13953811e-19 1.56384239e-32 1.74527155e-02 2.87463348e-08
 1.27420448e-19 2.70868440e-26 1.15939592e-33 3.40992353e-10
 2.01781822e-14 1.92274395e-21 4.30275578e-25 2.15317583e-01
 2.74039630e-28 1.85259563e-25 1.08509562e-03 1.84486263e-07
 1.46964210e-10 7.24184739e-26 1.88422330e-07 2.63071611e-23
 1.03446785e-37 4.91539560e-02 4.96809435e-18 2.39147958e-04]
margins:  [21.2958793   1.73965771 75.82068621 44.06099059 65.10717141
3.35914488
 32.33111087  0.61692895 22.24665724 58.65291919 21.56328119 39.91842177
 39.81300447 18.37916197 25.08727005 83.70779083 20.5955827  16.81765213
 53.97088168 67.62216697 11.19741372 30.20715651 13.50494123 85.68414174
 66.25327002 57.2421192   0.82831706 68.16630575 43.17541468  5.75865961
 69.12202693  3.57156341 24.84864198 35.87423153 33.13234697 56.5571617
 25.63761521 57.23390244 29.62254654  3.4051507  72.90563194 71.19320875
 47.24712037 17.09232265 61.25872129 17.23645227 24.43992306 43.26072939
 52.74154051 29.33900581 56.3492875   2.56400467 51.10048709 12.69077137
 13.14878084 71.48682364 40.47258304 14.66747023 30.20685518 48.79345784
 41.59225439  8.97499537 75.54325855 26.82719084  8.81249701 14.73880713
  4.17045505  8.09687555 58.78737821 15.59270513 56.39225061 13.97023672
 50.88140666 16.1473454  14.010979   70.92252069 41.53650692 73.23557711
  4.03065321 17.36475554 43.50679472 58.87074936 75.83740896 21.79916106
 31.53417447 47.70053365 56.10537163  1.293165   63.46428006 56.94803963
  6.82500148 15.50569065 22.64083203 57.88733608 15.48457977 51.99220105
 85.1617613   2.96239483 39.84349534  8.33818895]
```

```
Finished 70000 iterations
theta:  [-107.04156569  107.25200975  107.08020705]
theta gap:  0.0009315199306059814
grad:  [ 5.37193268e-05 -5.39154728e-05 -5.37089335e-05]
probs:  [1.72402427e-10 1.37675204e-01 1.69882549e-35 6.27505832e-21
 1.39624468e-30 2.81013123e-02 1.50245139e-15 3.42253892e-01
 6.26376746e-11 1.27423260e-27 1.29046754e-10 4.96167657e-19
 5.55229367e-19 3.71664737e-09 3.14638510e-12 4.13259993e-39
 3.59810159e-10 1.93628439e-08 1.78778395e-25 9.80706483e-32
 7.33651875e-06 1.40914593e-14 6.41164905e-07 5.09251948e-40
 4.17212472e-31 5.64439055e-27 2.94714092e-01 5.50047957e-32
 1.59840298e-20 2.28528544e-03 2.00681664e-32 2.23754695e-02
 4.03414241e-12 3.54292871e-17 6.38387292e-16 1.15659770e-26
 1.74902072e-12 5.68740705e-27 2.60702702e-14 2.65381695e-02
 3.69215274e-34 2.26421020e-33 2.14846155e-22 1.44295097e-08
 8.07235315e-29 1.24301703e-08 6.24316429e-12 1.45768733e-20
 6.54602900e-25 3.52547841e-14 1.44181835e-26 6.27317448e-02
 3.69529472e-24 1.50952632e-06 9.28998332e-07 1.64811382e-33
 2.74516196e-19 1.88009060e-07 1.41071673e-14 4.19631092e-23
 8.42846320e-20 7.62181153e-05 2.27899122e-35 4.96216601e-13
 9.11886739e-05 1.74228306e-07 1.21023869e-02 1.93800685e-04
 1.10032356e-27 7.04074196e-08 1.37718799e-26 3.91118296e-07
 4.66301809e-24 3.92849052e-08 3.75545588e-07 3.01362622e-33
 8.93140545e-20 2.62119015e-34 1.39649646e-02 1.08726192e-08
 1.11814068e-20 1.00683536e-27 1.67863983e-35 1.00761565e-10
 3.47555288e-15 1.34193395e-22 1.86495665e-26 2.03137556e-01
 7.89052820e-30 7.65982378e-27 7.39411619e-04 7.72378855e-08
 4.14867225e-11 2.86087247e-27 7.88352995e-08 1.43502652e-24
 8.89070235e-40 4.20313184e-02 5.34057798e-19 1.51058295e-04]
margins:  [22.48118968  1.83473467 80.06054113 46.51770417 68.74376653
3.54343529
 34.13167836  0.65326617 23.49365419 61.9274534  22.77084634 42.14737306
 42.03490565 19.41044382 26.48476691 88.38191189 21.74544456 17.75990985
 56.98365049 71.39961995 11.82263878 31.8932075  14.25997851 90.47563103
 69.95171245 60.43913528  0.87259756 71.97788769 45.58269687  6.07897644
 72.98617327  3.77716043 26.23622738 37.87899288 34.98758653 59.72172974
 27.07196507 60.43154307 31.2779808   3.60227456 76.98168347 75.16808207
 49.89212002 18.05399043 64.68652267 18.20313922 25.79953396 45.6748507
 55.68576872 30.97617515 59.50130736  2.70410192 53.95498192 13.40371314
 13.88915796 75.48567657 42.73927669 15.48677549 31.89209341 51.52525135
 43.92008741  9.48183517 79.76674536 28.33176387  9.30248867 15.56289912
  4.40217637  8.5484865  62.07419323 16.46896712 59.54716869 14.75425539
 53.72237933 17.05242544 14.79488559 74.88216399 43.86212809 77.32426469
  4.25714022 18.3370182  45.94003466 62.16298541 80.07249442 23.01826413
 33.29302283 50.36276023 59.24397461  1.36679871 67.01188971 60.13380853
  7.20891611 16.37637568 23.90564768 61.11867087 16.3559049  54.9008589
 89.91839767  3.12640007 42.07378288  8.79769367]
```

```
Finished 80000 iterations
theta:  [-112.16638881  112.39737225  112.20335022]
theta gap:  0.0008490730225995943
grad:  [ 4.89573832e-05 -4.91710659e-05 -4.89349658e-05]
probs:  [5.90885224e-11 1.27869606e-01 3.65306878e-37 6.80587219e-22
 5.19372380e-32 2.39226889e-02 2.95248836e-16 3.34613778e-01
 2.02208978e-11 6.58311169e-29 4.32015621e-11 6.60066626e-20
 7.43821653e-20 1.45825555e-09 8.89765904e-13 6.01538272e-41
 1.27157576e-10 8.24244133e-09 1.17069015e-26 3.21073470e-33
 4.16567804e-06 3.06487039e-15 3.23701618e-07 6.64444300e-42
 1.46913517e-32 3.12589687e-28 2.86717815e-01 1.74345835e-33
 1.81290034e-21 1.71277997e-03 6.06799685e-34 1.85929675e-02
 1.14959247e-12 5.77120324e-18 1.18830459e-16 6.57928808e-28
 4.77078204e-13 3.14614256e-28 5.82330294e-15 2.22245461e-02
 9.21291042e-36 6.20658775e-35 1.95541704e-23 6.02251436e-09
 3.61733657e-30 5.17278530e-09 1.82875455e-12 1.64130452e-21
 4.56054097e-26 8.01635065e-15 8.29953547e-28 5.57506898e-02
 2.78943364e-25 7.90292658e-07 4.73973810e-07 4.40532886e-35
 3.51846859e-20 8.95870964e-08 3.07183310e-15 3.52994698e-24
 1.02287761e-20 4.80502553e-05 4.97605229e-37 1.26765027e-13
 5.85822806e-05 8.25857289e-08 9.83916056e-03 1.28800121e-04
 5.61046932e-29 3.17753271e-08 7.90465998e-28 1.92046791e-07
 3.56568691e-25 1.72927918e-08 1.84559712e-07 8.37600770e-35
 1.08557037e-20 6.48275948e-36 1.13954466e-02 4.50561560e-09
 1.23450147e-21 5.10659637e-29 3.63439015e-37 3.33942528e-11
 7.08222749e-16 1.20700880e-23 1.08652094e-27 1.92486803e-01
 3.17818049e-31 4.27589397e-28 5.21880919e-04 3.50475840e-08
 1.32078232e-11 1.53730768e-28 3.57174449e-08 1.02957060e-25
 1.20017559e-41 3.64805595e-02 7.07903049e-20 9.99712916e-05]
margins:  [23.55198442  1.9199279  83.90008087 48.73908625 72.03527204
3.70871446
 35.75871316  0.68739069 24.6243045  64.89046016 23.86514446 44.16453127
 44.04507075 20.34602494 27.747818   92.61166884 22.78559404 18.61396925
 59.70961897 74.81880828 12.38862733 33.41877111 14.94344335 94.81479304
 73.29804906 63.33266136  0.9113786  75.42943737 47.75935899  6.36792327
 76.48486462  3.96620387 27.49161361 39.69365108 36.66883391 62.58845606
 28.37109597 63.32620549 32.77690878  3.78408269 80.67245754 78.76486699
 52.28885365 18.92776099 67.79181479 19.07985455 27.02738595 47.85879559
 58.34977117 32.45729311 62.35618306  2.82950045 56.53878875 14.05086172
 14.5621133  79.10766334 44.79367602 16.22805445 33.41650191 54.00075938
 46.02908201  9.94321506 83.59101158 29.6964412   9.7450197  16.30942886
  4.611497    8.95712    65.05033332 17.26457569 62.40493015 15.4655266
 56.29327061 17.87297606 15.5052926  78.46510686 45.96959633 81.02391708
  4.46308055 19.2179413  48.14361973 65.14443459 83.90520711 24.1226373
 34.88377301 52.7713119  62.08681672  1.4339318  70.22382902 63.01938941
  7.5575491  17.16655912 25.05021179 64.04234998 17.14762658 57.5354855
 94.22352094  3.27381316 44.0945649   9.21052752]
```

```
Finished 90000 iterations
theta:  [-116.86340448  117.11642203  116.89769046]
theta gap:  0.0007824983080599451
grad:  [ 4.51121714e-05 -4.53384732e-05 -4.50816023e-05]
probs:  [2.21829225e-11 1.19494523e-01 1.08026300e-38 8.90130040e-23
 2.54268085e-33 2.06559997e-02 6.66054621e-17 3.27483225e-01
 7.16128517e-12 4.35801533e-30 1.58268666e-11 1.03936830e-20
 1.17937231e-20 6.17463874e-10 2.80019834e-13 1.24705080e-42
 4.90312040e-11 3.76373740e-09 9.63153300e-28 1.39805387e-34
 2.47965821e-06 7.57450326e-16 1.72989363e-07 1.24259903e-43
 6.84628326e-34 2.20418722e-29 2.79699693e-01 7.35959670e-35
 2.47023110e-22 1.31571928e-03 2.45401047e-35 1.56428943e-02
 3.63890929e-13 1.09353870e-18 2.54050836e-17 4.74391178e-29
 1.44911110e-13 2.21512209e-29 1.47344219e-15 1.88248564e-02
 3.12367439e-37 2.29860227e-36 2.16863831e-24 2.69610069e-09
 2.09542545e-31 2.31247191e-09 5.94841880e-13 2.21955997e-22
 3.97110331e-27 2.06387136e-15 6.05206271e-29 5.00585718e-02
 2.61199028e-26 4.36000203e-07 2.55173230e-07 1.58944098e-36
 5.34005792e-21 4.54184321e-08 7.60235271e-16 3.64151455e-25
 1.47756480e-21 3.14104662e-05 1.49279035e-38 3.62006469e-14
 3.90839277e-05 4.16438254e-08 8.13885873e-03 8.85834672e-05
 3.66302563e-30 1.52920753e-08 5.74766085e-29 9.99206117e-08
 3.38048893e-26 8.14173989e-09 9.61774680e-08 3.14179809e-36
 1.56971214e-21 2.18511891e-37 9.44738301e-03 2.00783611e-09
 1.63625074e-22 3.31677808e-30 1.08388700e-38 1.21272217e-11
 1.64972853e-16 1.32850959e-24 8.01011696e-29 1.83030859e-01
 1.67203076e-32 3.03108274e-29 3.78801564e-04 1.69561666e-08
 4.62665690e-12 1.05524617e-29 1.72424146e-08 9.18695844e-27
 2.32072641e-43 3.20409731e-02 1.10862249e-20 6.86560052e-05]
margins:  [24.53169838  1.99722561 87.421029   50.77325976 75.05208909
3.85887714
 37.24774509  0.71959022 25.66233166 67.60553604 24.8693122  46.01308873
 45.8867195  21.20540055 28.90391596 96.4877925  23.7385642  19.39785338
 62.2073402  77.95281198 12.90738734 34.81657372 15.57003556 98.79395382
 76.36418725 65.98460886  0.94595172 78.59447312 49.75256034  6.63205519
 79.69275464  4.14197197 28.64192222 41.35711272 38.21158238 65.21810563
 29.56265587 65.97966018 34.1511751   3.95357283 84.05663844 82.06076212
 54.48794277 19.73145929 70.64038127 19.88494879 28.15048077 49.85956308
 60.79075354 33.81419287 64.97456854  2.94320657 58.90709993 14.64562269
 15.18132294 82.42968098 46.67905045 16.90734778 34.81290372 56.27222765
 47.96389163 10.36833799 87.09758645 30.9496994  10.14976015 16.99411269
  4.80293315  9.33147673 67.77926331 17.99593108 65.02617473 16.11888975
 58.64919206 18.62626193 16.15707063 81.74826807 47.9033947  84.41397819
  4.65252521 20.02620826 50.16446455 67.87855894 87.41767988 25.13556847
 36.34075074 54.97798453 64.69426233  1.49594656 73.16868406 65.6660478
  7.87811919 17.89263424 26.09918656 66.72119362 17.87589351 59.95201259
 98.16927875  3.40817427 45.94858362  9.5863333 ]
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-11-4a6d52a50c67> in <module>
----> 1 logistic_regression_print(Xb, Yb)

<ipython-input-9-1ebd84199d22> in logistic_regression_print(X, Y)
     25         grad, probs, margins = calc_grad(X, Y, theta)
     26         theta = theta - learning_rate * grad
---> 27         theta_gap = np.linalg.norm(prev_theta - theta)
     28         if i % 10000 == 0:
     29             print('Finished %d iterations' % i)

KeyboardInterrupt:
```

Message shows, compared to A, B has: 1. bigger and margins 2. smaller prob 3. bigger grad 4. increasing theta 5. every margin is positive

In this logistic regression, we have:

$$\hat{\gamma}^{(i)} = y^{(i)}(\theta^T x^{(i)} + b) \tag{1}$$

$$p^{(i)} = \frac{1}{1 + \exp(\hat{\gamma}^{(i)})} \tag{2}$$

$$\nabla = -\frac{1}{m}\sum_{i=1}^{m} y^{(i)} x^{(i)} p^{(i)} \tag{3}$$

$$\theta := \theta - \alpha\nabla \tag{4}$$

It means what we want is:

$$\arg\max_{\theta} \hat{\gamma} \tag{5}$$

In fact, our program will stop automatically only if $\alpha\nabla$ is small enough. Actually, the learning rate, α, which in our program is too large to automatically stop.

But this dosen't explain why A can stop automatically and B can't (they have same α), so the real different is in the ∇.

In the case of B:

Because every margin is positive, means:

$$\hat{\gamma}^{(i)} \geq 0 \text{ for every } i$$

So, the only way to increase $\hat{\gamma}$ is to increase $\parallel \theta \parallel$ and $\parallel b \parallel$, this led to an unlimited increase in theta.

In the case of A:

Observe the image of data set A, we can findout no matter how hard the algorithm works on A, it will always misclassify some points. What's more, under the best predictions that the algorithm can make, both some positive and negative points are misclassified.

Consider (2), misclassified points have larger $ p^{(i)} $, so they have a greater weight in gradient ∇. It turns out that our decision boundary is controlled by those misclassified points, and algorithm will try to find the boundary between these points.

Misclassified points also help algorithm stop increasing theta. When algorithm try to increase $\| \theta \|$ and $\| b \|$, in the case of these misclassified points, $\hat{\gamma}^{(i)}$ will keep decreasing, this contradicts (5). In the other hand, because misclassified points have a greater weight in gradient, our algorithm is more likely to increase $\hat{\gamma}^{(i)}$ of these points ($\hat{\gamma}^{(i)} < 0$), rather than increase $\| \theta \|$ and $\| b \|$.

3 (c)

[4]:
```
import random
```

[59]:
```
def logistic_regression_justify(X, Y,
                                constant_learning_rate=10,
                                decreasing_learnging_rate=False,
                                linear_scaling_input=False,
                                regular_loss=False,
                                noise=False):
    '''
    A modifed version of logsitic regression
    only to justify my answers
    '''

    def calc_grad(X, Y, theta):
        m, n = X.shape

        margins = Y * X.dot(theta)
        probs = 1. / (1 + np.exp(margins))
        grad = -(1./m) * (X.T.dot(probs * Y))

        return grad

    def plot_data(x, y, title=None):
        plt.figure()
        plt.plot(x[y == 1, -2], x[y == 1, -1], 'go', linewidth=2)
        plt.plot(x[y == -1, -2], x[y == -1, -1], 'bx', linewidth=2)
        plt.xlabel('x1')
        plt.ylabel('x2')
        if not title == None: plt.title(title)

    # linear scaling input
```

```
    if linear_scaling_input:
        for xi in X:
            xi[1:] *= 0.5

    # gaussian noise
    if noise:
        mu = 0
        sigma = 0.05
        for xi in X:
            xi[-1] += random.gauss(mu, sigma)
            xi[-2] += random.gauss(mu, sigma)
        plot_data(X, Y, title='noise')

    m, n = X.shape
    theta = np.zeros(n)

    # constant learning rate
    learning_rate = constant_learning_rate

    i = 0
    while True:
        i += 1
        prev_theta = theta
        grad = calc_grad(X, Y, theta)

        # decreasing learning rate
        if decreasing_learnging_rate:
            learning_rate = 1 / i**3 * constant_learning_rate

        theta = theta - learning_rate * grad

        if i % 10000 == 0:
            print('Finished %d iterations' % i)
        if np.linalg.norm(prev_theta - theta) < 1e-15:
            print('Converged in %d iterations' % i)
            break
    return
```

3.1 i

No (or yes?).

Consider (4), if we want our algorithm stop automatically, $\alpha\nabla$ should small enough.

In the case of B, although ∇ is large, we still can reduce α to achieve it. But a constant learning rate just a scalar of ∇.

So, if we have a small α, in some very lucky situations, it may just exist a small enough ∇, resulting in $\alpha\nabla$ being small enough to stop the algorithm...(I guess)

[65]:
```
logistic_regression_justify(Xb, Yb, constant_learning_rate=0.1)
```

```
Finished 10000 iterations
Finished 20000 iterations
Finished 30000 iterations
Finished 40000 iterations
Finished 50000 iterations
Finished 60000 iterations
Finished 70000 iterations
Finished 80000 iterations
Finished 90000 iterations
Finished 100000 iterations
Finished 110000 iterations
Finished 120000 iterations
Finished 130000 iterations
Finished 140000 iterations
Finished 150000 iterations
Finished 160000 iterations
Finished 170000 iterations
Finished 180000 iterations
Finished 190000 iterations
Finished 200000 iterations
Finished 210000 iterations
Finished 220000 iterations
```

```
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-65-7611a065c5f1> in <module>
----> 1 logistic_regression_justify(Xb, Yb, constant_learning_rate=0.1)

<ipython-input-59-154909c08718> in logistic_regression_justify(X, Y,␣
↪constant_learning_rate, decreasing_learnging_rate, linear_scaling_input,␣
↪regular_loss, noise)
     67         if i % 10000 == 0:
     68             print('Finished %d iterations' % i)
---> 69         if np.linalg.norm(prev_theta - theta) < 1e-15:
     70             print('Converged in %d iterations' % i)
     71             break

<__array_function__ internals> in norm(*args, **kwargs)

~/anaconda3/envs/cs229/lib/python3.9/site-packages/numpy/linalg/linalg.py in␣
↪norm(x, ord, axis, keepdims)
   2528                 sqnorm = dot(x.real, x.real) + dot(x.imag, x.imag)
   2529             else:
-> 2530                 sqnorm = dot(x, x)
   2531             ret = sqrt(sqnorm)
   2532             if keepdims:
```

```
<__array_function__ internals> in dot(*args, **kwargs)

KeyboardInterrupt:
```

3.2 ii

Yes.

If we keep decreasing learning rate faster than theta, eventually we will get a learning rate that makes $\alpha\nabla$ small enough to stop algorithm.

[55]:
```
logistic_regression_justify(Xb, Yb, decreasing_learnging_rate=True)
```

```
Finished 10000 iterations
Finished 20000 iterations
Finished 30000 iterations
Finished 40000 iterations
Finished 50000 iterations
Finished 60000 iterations
Finished 70000 iterations
Finished 80000 iterations
Finished 90000 iterations
Finished 100000 iterations
Finished 110000 iterations
Finished 120000 iterations
Converged in 122421 iterations
```

3.3 iii

No (or yes?).

Consider (1), scaling input feature just another way to scaling theta.

But, consider (4), small input feature can help ∇ become smaller, then $\alpha\nabla$ may small enough to stop algorithm.

Obviously, this isn't a good way to solve this problem …

[66]:
```
logistic_regression_justify(Xb, Yb, linear_scaling_input=True)
```

```
Finished 10000 iterations
Finished 20000 iterations
Finished 30000 iterations
Finished 40000 iterations
Finished 50000 iterations
Finished 60000 iterations
Finished 70000 iterations
Finished 80000 iterations
Finished 90000 iterations
```

```
Finished 100000 iterations
Finished 110000 iterations
Finished 120000 iterations
Finished 130000 iterations
Finished 140000 iterations
Finished 150000 iterations
Finished 160000 iterations
Converged in 164349 iterations
```

3.4 iv

Yes.

Adding a regularization term is an effective way to avoid unlimited increasing of theta. So this modification can let algorithm stop automatically.

3.5 v

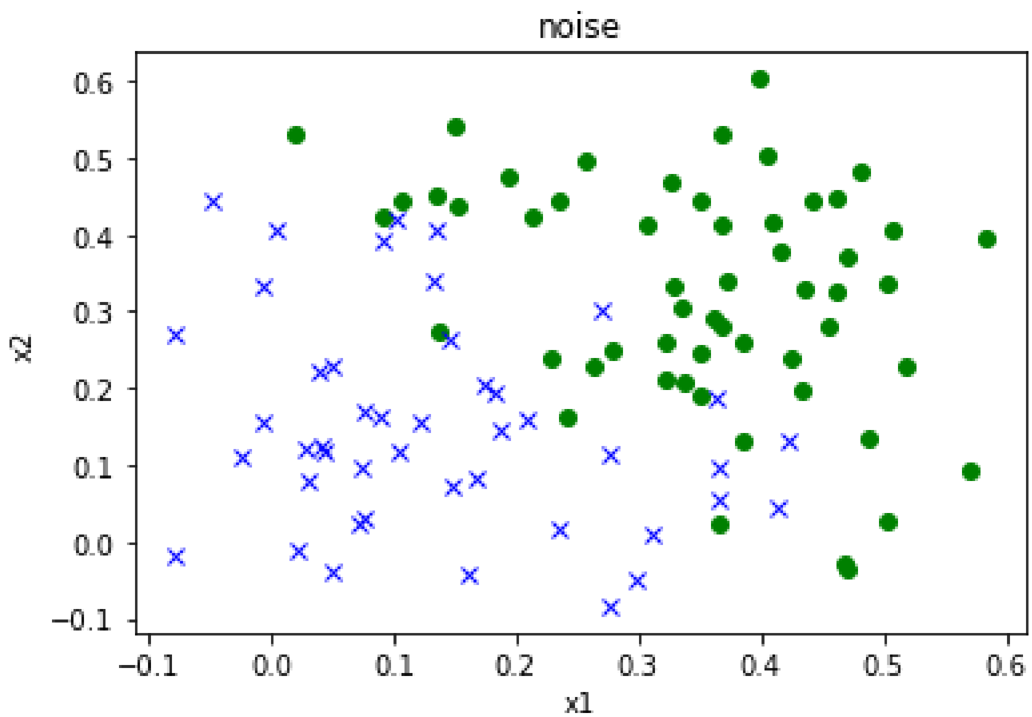
Yes.

Add zero-mean Gaussian noise will make the data set B similar to the data set A, there are always some points that algorithm will misclassify.

So, simillar to the case of A, program will stop automatically.

[58]:
```
logistic_regression_justify(Xb, Yb, noise=True)
```

```
Finished 10000 iterations
Converged in 16033 iterations
```



4 (d)

No.

Hinge loss:

$$L(y) = \max(0, 1 - \hat{y}y)$$

if a point i is correctly classified, then $L(y^{(i)}) = 0$, and algorithm will stop on it, instead of increasing θ.

```
[ ]:
```