Homework 2 of Artificial Intelligent

Jianqiang Li

• Do multiple starting points help in finding better solutions?

Yes, trying different starting points of parameters can help to find better solutions. Estimation is sensitive to the starting point.

EM can reach a local maximum in likelihood, so some other starting points can let to converge and have almost the same solutions, but the algorithm needs to take different number of iteration rounds to converge with different starting points.

If the starting points set 0.5 for all parameters of 100% missing data, then the solutions cannot be good enough.

• Do some of the different solutions have the same likelihood scores?

Yes, some different solutions have almost the same likelihood scores.

EM resembles a gradient-based hill-climbing algorithm, and EM can reach a local maximum in likelihood.

• How does the data missing rate affect your algorithm and the results?

With starting parameters as below:

P(gender=M)=0.7; P(weight=greater_than_130|gender=M)=0.8;

P(weight=greater_than_130|gender=F)=0.4; P(height=greater_than_55|gender=M)=0.7;

P(height= greater_than_55|gender=F)=0.3;

Converge when change of log likelihood value < 0.001

We can know all can be converge with different data missing rate. And the iteration rounds seem not related to the data missing rate. 50% missing rate need the most iteration rounds. Picking a good starting point can converge fast and have a better solution.

Data missing rate	Iteration round	Converge value	Log likelihood scores
10%	4	0.00016	-12.799
30%	3	0.00073	-14.750
50%	17	0.00083	-12.355
70%	4	0.00082	-12.912
100%	5	0.00049	-10.749

Experimental Results report

With starting point:

P(gender=M)=0.7; P(weight=greater_than_130|gender=M)=0.8;

P(weight=greater_than_130|gender=F)=0.4; P(height= greater_than_55|gender=M)=0.7;

P(height= greater_than_55|gender=F)=0.3;

Below are results with different data missing rate:

1. hw2dataset_10

File name: hw2dataset_10.txt

Starting point of the parameters table.

P(G=0): 0.7

P(G=1): 0.30000000000000004

P(W=0/G=0): 0.8

P(W=1/G=0): 0.19999999999999996

P(W=0/G=1): 0.4

P(W=1/G=1): 0.6

P(H=0/G=0): 0.7

P(H=1/G=0): 0.30000000000000004

P(H=0/G=1): 0.3

P(H=1/G=1): 0.7

Parameters table at iteration No.1

P(G=0): 0.7291277258566977

P(G=1): 0.27087227414330217

P(W=0/G=0): 0.8628498184148686

P(W=1/G=0): 0.13715018158513137

P(W=0/G=1): 0.6308223116733755

P(W=1/G=1): 0.36917768832662445

P(H=0/G=0): 0.6799829096346933

P(H=1/G=0): 0.3200170903653067

P(H=0/G=1): 0.015526164462334678

P(H=1/G=1): 0.9844738355376653

Log likelihood: -12.835066497653445

Parameters table at iteration No.2

P(G=0): 0.7269318737480295

P(G=1): 0.27306812625197047

P(W=0/G=0): 0.8624355271637708
P(W=1/G=0): 0.1375644728362292
P(W=0/G=1): 0.6337910199459671
P(W=1/G=1): 0.3662089800540329
P(H=0/G=0): 0.6873984379622053
P(H=1/G=0): 0.31260156203779466
P(H=0/G=1): 0.0011285296636236173
P(H=1/G=1): 0.9988714703363764

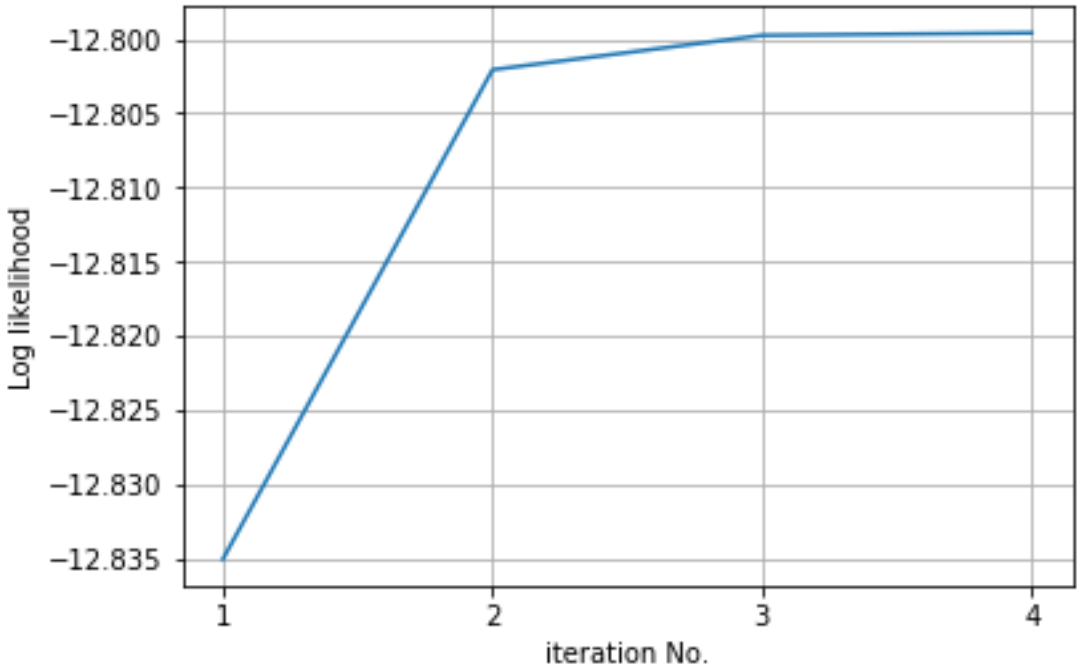
Log likelihood: -12.802056745774902
Parameters table at iteration No.3

P(G=0): 0.7265434944746942
P(G=1): 0.2734565055253058
P(W=0/G=0): 0.8623619910432175
P(W=1/G=0): 0.13763800895678246
P(W=0/G=1): 0.6343111318273393
P(W=1/G=1): 0.36568886817266066
P(H=0/G=0): 0.6881588693301972
P(H=1/G=0): 0.3118411306698028
P(H=0/G=1): 8.28297101235954e-05
P(H=1/G=1): 0.9999171702898764

Log likelihood: -12.799738844363583
Parameters table at iteration No.4

P(G=0): 0.7264854852204777
P(G=1): 0.27351451477952227
P(W=0/G=0): 0.8623510007641082
P(W=1/G=0): 0.1376489992358918
P(W=0/G=1): 0.6343886901921489
P(W=1/G=1): 0.36561130980785106
P(H=0/G=0): 0.6882427028556031
P(H=1/G=0): 0.3117572971443969
P(H=0/G=1): 6.091327531325839e-06
P(H=1/G=1): 0.9999939086724686

Log likelihood: -12.79956969684856
change value at the final iteration (diff value): 0.00016914751502383751
Iteration round: 4



2. hw2dataset_30

File name: hw2dataset_30.txt

Starting point of the parameters table.

P(G=0): 0.7

P(G=1): 0.30000000000000004

P(W=0/G=0): 0.8

P(W=1/G=0): 0.19999999999999996

P(W=0/G=1): 0.4

P(W=1/G=1): 0.6

P(H=0/G=0): 0.7

P(H=1/G=0): 0.30000000000000004

P(H=0/G=1): 0.3

P(H=1/G=1): 0.7

Parameters table at iteration No.1

P(G=0): 0.5707943925233645

P(G=1): 0.42920560747663555

P(W=0/G=0): 0.8686041751944331

P(W=1/G=0): 0.13139582480556689

P(W=0/G=1): 0.24278715296679368

P(W=1/G=1): 0.7572128470332063

P(H=0/G=0): 0.5182153090462546
P(H=1/G=0): 0.48178469095374543
P(H=0/G=1): 0.24278715296679368
P(H=1/G=1): 0.7572128470332063

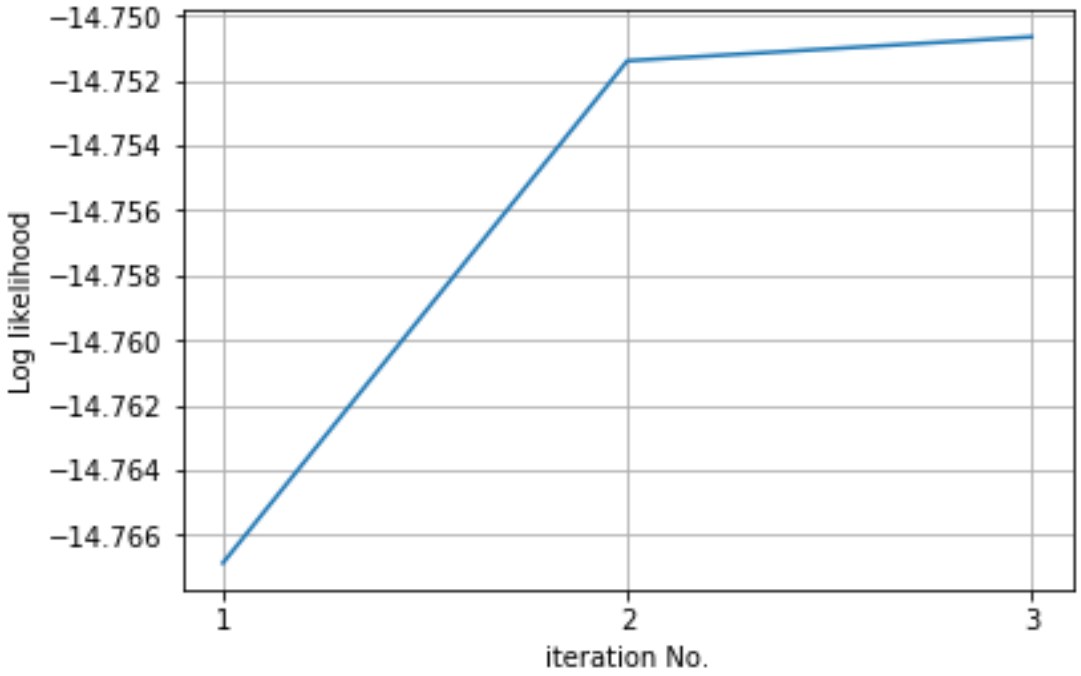
Log likelihood: -14.766867391601256
Parameters table at iteration No.2

P(G=0): 0.5583209171962877
P(G=1): 0.4416790828037122
P(W=0/G=0): 0.88751438814411
P(W=1/G=0): 0.11248561185589001
P(W=0/G=1): 0.23655671471522133
P(W=1/G=1): 0.7634432852847787
P(H=0/G=0): 0.5292974669434192
P(H=1/G=0): 0.47070253305658083
P(H=0/G=1): 0.23655671471522133
P(H=1/G=1): 0.7634432852847787

Log likelihood: -14.751386199901907
Parameters table at iteration No.3

P(G=0): 0.5559944419936487
P(G=1): 0.4440055580063514
P(W=0/G=0): 0.891544827616126
P(W=1/G=0): 0.10845517238387403
P(W=0/G=1): 0.2349205526291099
P(W=1/G=1): 0.7650794473708901
P(H=0/G=0): 0.5318290015318705
P(H=1/G=0): 0.4681709984681295
P(H=0/G=1): 0.2349205526291099
P(H=1/G=1): 0.7650794473708901

Log likelihood: -14.750649031117302
change value at the final iteration (diff value): 0.0007371687846049468
Iteration round: 3



3. hw2dataset_50

File name: hw2dataset_50.txt

Starting point of the parameters table.

P(G=0): 0.7

P(G=1): 0.30000000000000004

P(W=0/G=0): 0.8

P(W=1/G=0): 0.19999999999999996

P(W=0/G=1): 0.4

P(W=1/G=1): 0.6

P(H=0/G=0): 0.7

P(H=1/G=0): 0.30000000000000004

P(H=0/G=1): 0.3

P(H=1/G=1): 0.7

Parameters table at iteration No.1

P(G=0): 0.6631435481226431

P(G=1): 0.3368564518773569

P(W=0/G=0): 0.7410658268456335

P(W=1/G=0): 0.2589341731543665

P(W=0/G=1): 0.3222944895016397

P(W=1/G=1): 0.6777055104983603

P(H=0/G=0): 0.7863710397730258
P(H=1/G=0): 0.2136289602269742
P(H=0/G=1): 0.23310557985872396
P(H=1/G=1): 0.7668944201412761

Log likelihood: -12.554274669141389
Parameters table at iteration No.2

P(G=0): 0.6622503399260911
P(G=1): 0.33774966007390905
P(W=0/G=0): 0.7358722877465778
P(W=1/G=0): 0.26412771225342224
P(W=0/G=1): 0.3335853166306181
P(W=1/G=1): 0.6664146833693819
P(H=0/G=0): 0.8141287037177797
P(H=1/G=0): 0.18587129628222032
P(H=0/G=1): 0.18014226633417907
P(H=1/G=1): 0.8198577336658209

Log likelihood: -12.488786891551994
Parameters table at iteration No.3

P(G=0): 0.6666463894592818
P(G=1): 0.3333536105407181
P(W=0/G=0): 0.7273878647408981
P(W=1/G=0): 0.2726121352591019
P(W=0/G=1): 0.3452475167354636
P(W=1/G=1): 0.6547524832645364
P(H=0/G=0): 0.8282598822685177
P(H=1/G=0): 0.17174011773148234
P(H=0/G=1): 0.14352188918583472
P(H=1/G=1): 0.8564781108141652

Log likelihood: -12.452983772437014
Parameters table at iteration No.4

P(G=0): 0.6724126425179263
P(G=1): 0.32758735748207357
P(W=0/G=0): 0.7197135211323213
P(W=1/G=0): 0.28028647886767866
P(W=0/G=1): 0.35427352963670206
P(W=1/G=1): 0.6457264703632979
P(H=0/G=0): 0.8362576130191468
P(H=1/G=0): 0.1637423869808532

P(H=0/G=1): 0.11505269584808235
P(H=1/G=1): 0.8849473041519177

Log likelihood: -12.428790812653807
Parameters table at iteration No.5

P(G=0): 0.6780591622519272
P(G=1): 0.3219408377480729
P(W=0/G=0): 0.7137591103041983
P(W=1/G=0): 0.28624088969580175
P(W=0/G=1): 0.3604050248146845
P(W=1/G=1): 0.6395949751853155
P(H=0/G=0): 0.8411564887743912
P(H=1/G=0): 0.1588435112256088
P(H=0/G=1): 0.09208566426749096
P(H=1/G=1): 0.907914335732509

Log likelihood: -12.411177077762005
Parameters table at iteration No.6

P(G=0): 0.6830635037968642
P(G=1): 0.31693649620313585
P(W=0/G=0): 0.7093277704011578
P(W=1/G=0): 0.2906722295988422
P(W=0/G=1): 0.3643761106494648
P(W=1/G=1): 0.6356238893505353
P(H=0/G=0): 0.8443121662059746
P(H=1/G=0): 0.1556878337940254
P(H=0/G=1): 0.07345690298193025
P(H=1/G=1): 0.9265430970180697

Log likelihood: -12.397921379445048
Parameters table at iteration No.7

P(G=0): 0.6872941776914515
P(G=1): 0.3127058223085485
P(W=0/G=0): 0.7060346431334527
P(W=1/G=0): 0.29396535686654734
P(W=0/G=1): 0.36694711879299496
P(W=1/G=1): 0.6330528812070051
P(H=0/G=0): 0.8464179575696799
P(H=1/G=0): 0.15358204243032014
P(H=0/G=1): 0.05839950703204322
P(H=1/G=1): 0.9416004929679568

Log likelihood: -12.387778130902426
Parameters table at iteration No.8

P(G=0): 0.6907772991410471
P(G=1): 0.30922270085895287
P(W=0/G=0): 0.7035614232803242
P(W=1/G=0): 0.2964385767196758
P(W=0/G=1): 0.36865255988622886
P(W=1/G=1): 0.6313474401137711
P(H=0/G=0): 0.8478647802101559
P(H=1/G=0): 0.15213521978984412
P(H=0/G=1): 0.04629109392631642
P(H=1/G=1): 0.9537089060736836

Log likelihood: -12.379949807985053
Parameters table at iteration No.9

P(G=0): 0.6935982055287266
P(G=1): 0.30640179447127347
P(W=0/G=0): 0.7016829692202016
P(W=1/G=0): 0.29831703077979843
P(W=0/G=1): 0.3698214362430025
P(W=1/G=1): 0.6301785637569974
P(H=0/G=0): 0.8488851571084199
P(H=1/G=0): 0.15111484289158006
P(H=0/G=1): 0.03660154258881239
P(H=1/G=1): 0.9633984574111876

Log likelihood: -12.37388010000845
Parameters table at iteration No.10

P(G=0): 0.6958582892670515
P(G=1): 0.30414171073294843
P(W=0/G=0): 0.7002437461349342
P(W=1/G=0): 0.29975625386506577
P(W=0/G=1): 0.3706482233986489
P(W=1/G=1): 0.6293517766013511
P(H=0/G=0): 0.849621638795185
P(H=1/G=0): 0.15037836120481496
P(H=0/G=1): 0.028880418547478713
P(H=1/G=1): 0.9711195814525213

Log likelihood: -12.369161174977371

Parameters table at iteration No.11

P(G=0): 0.6976559191001531

P(G=1): 0.3023440808998469

P(W=0/G=0): 0.699134282929774

P(W=1/G=0): 0.300865717070226

P(W=0/G=1): 0.3712486413960385

P(W=1/G=1): 0.6287513586039615

P(H=0/G=0): 0.8501637460083209

P(H=1/G=0): 0.1498362539916791

P(H=0/G=1): 0.022749677693257846

P(H=1/G=1): 0.9772503223067421

Log likelihood: -12.365486062369015

Parameters table at iteration No.12

P(G=0): 0.6990786276625399

P(G=1): 0.3009213723374601

P(W=0/G=0): 0.6982754353841746

P(W=1/G=0): 0.3017245646158254

P(W=0/G=1): 0.3716936621428696

P(W=1/G=1): 0.6283063378571304

P(H=0/G=0): 0.8505691505643378

P(H=1/G=0): 0.1494308494356622

P(H=0/G=1): 0.017895988741375227

P(H=1/G=1): 0.9821040112586248

Log likelihood: -12.36262046572916

Parameters table at iteration No.13

P(G=0): 0.7002007639013065

P(G=1): 0.2997992360986935

P(W=0/G=0): 0.6976086675227869

P(W=1/G=0): 0.30239133247721306

P(W=0/G=1): 0.37202856000492046

P(W=1/G=1): 0.6279714399950795

P(H=0/G=0): 0.8508760701485933

P(H=1/G=0): 0.14912392985140666

P(H=0/G=1): 0.014062496460883365

P(H=1/G=1): 0.9859375035391167

Log likelihood: -12.360384153357678

Parameters table at iteration No.14

P(G=0): 0.7010837400924584
P(G=1): 0.2989162599075416
P(W=0/G=0): 0.6970899655307427
P(W=1/G=0): 0.30291003446925735
P(W=0/G=1): 0.37228339408235867
P(W=1/G=1): 0.6277166059176413
P(H=0/G=0): 0.8511105832519271
P(H=1/G=0): 0.14888941674807288
P(H=0/G=1): 0.011040580604367602
P(H=1/G=1): 0.9889594193956324

Log likelihood: -12.358637818675541
Parameters table at iteration No.15

P(G=0): 0.701777392040978
P(G=1): 0.2982226079590218
P(W=0/G=0): 0.6966858568075466
P(W=1/G=0): 0.3033141431924534
P(W=0/G=1): 0.3724788576492803
P(W=1/G=1): 0.6275211423507197
P(H=0/G=0): 0.8512909902691648
P(H=1/G=0): 0.14870900973083523
P(H=0/G=1): 0.008662082994320851
P(H=1/G=1): 0.9913379170056792

Log likelihood: -12.357273446211362
Parameters table at iteration No.16

P(G=0): 0.7023216948274694
P(G=1): 0.2976783051725306
P(W=0/G=0): 0.6963706860634704
P(W=1/G=0): 0.3036293139365296
P(W=0/G=1): 0.3726296394742164
P(W=1/G=1): 0.6273703605257837
P(H=0/G=0): 0.8514304567351839
P(H=1/G=0): 0.14856954326481608
P(H=0/G=1): 0.00679229410704728
P(H=1/G=1): 0.9932077058929527

Log likelihood: -12.356207096334122
Parameters table at iteration No.17

P(G=0): 0.7027484665497734
P(G=1): 0.2972515334502266

P(W=0/G=0): 0.6961246824189542

P(W=1/G=0): 0.3038753175810458

P(W=0/G=1): 0.3727464266258688

P(W=1/G=1): 0.6272535733741311

P(H=0/G=0): 0.8515386529556846

P(H=1/G=0): 0.14846134704431535

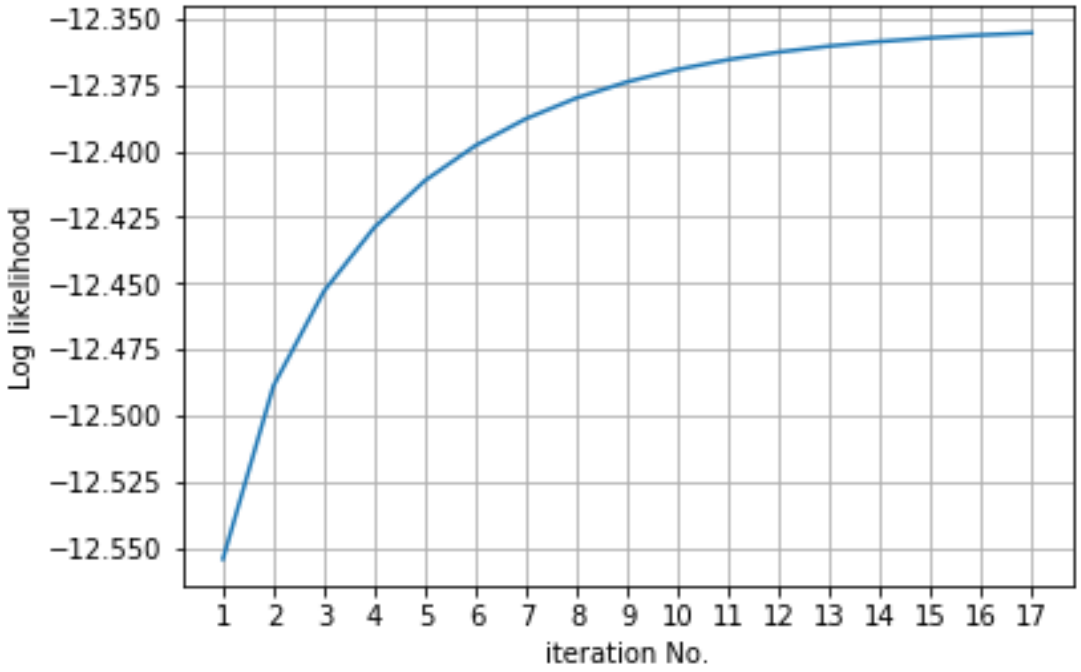
P(H=0/G=1): 0.005323832678554916

P(H=1/G=1): 0.994676167321445

Log likelihood: -12.355373432474241

change value at the final iteration (diff value): 0.0008336638598809287

Iteration round: 17



4. hw2dataset_70

File name: hw2dataset_70.txt

Starting point of the parameters table.

P(G=0): 0.7

P(G=1): 0.30000000000000004

P(W=0/G=0): 0.8

P(W=1/G=0): 0.19999999999999996

P(W=0/G=1): 0.4

P(W=1/G=1): 0.6

P(H=0/G=0): 0.7

P(H=1/G=0): 0.30000000000000004
P(H=0/G=1): 0.3
P(H=1/G=1): 0.7

Parameters table at iteration No.1

P(G=0): 0.5185624692572552
P(G=1): 0.4814375307427448
P(W=0/G=0): 0.4658817391716763
P(W=1/G=0): 0.5341182608283237
P(W=0/G=1): 0.12132667526596082
P(W=1/G=1): 0.8786733247340391
P(H=0/G=0): 0.5902133058253003
P(H=1/G=0): 0.4097866941746997
P(H=0/G=1): 0.19511883628561577
P(H=1/G=1): 0.8048811637143842

Log likelihood: -12.915781008718474
Parameters table at iteration No.2

P(G=0): 0.5232519570393851
P(G=1): 0.4767480429606149
P(W=0/G=0): 0.4668899299207624
P(W=1/G=0): 0.5331100700792376
P(W=0/G=1): 0.11683095771319423
P(W=1/G=1): 0.8831690422868058
P(H=0/G=0): 0.5906284472564612
P(H=1/G=0): 0.40937155274353876
P(H=0/G=1): 0.19077689029430814
P(H=1/G=1): 0.8092231097056919

Log likelihood: -12.914034573364605
Parameters table at iteration No.3

P(G=0): 0.5265207605623056
P(G=1): 0.47347923943769443
P(W=0/G=0): 0.4682442363911601
P(W=1/G=0): 0.5317557636088399
P(W=0/G=1): 0.11290819971304193
P(W=1/G=1): 0.8870918002869581
P(H=0/G=0): 0.5894650367115918
P(H=1/G=0): 0.4105349632884082
P(H=0/G=1): 0.1893101388187061
P(H=1/G=1): 0.8106898611812939

Log likelihood: -12.913008064732473

Parameters table at iteration No.4

P(G=0): 0.5290957643690666

P(G=1): 0.4709042356309334

P(W=0/G=0): 0.4697014126551441

P(W=1/G=0): 0.5302985873448559

P(W=0/G=1): 0.10932790183341105

P(W=1/G=1): 0.890672098166589

P(H=0/G=0): 0.5877488267947854

P(H=1/G=0): 0.41225117320521465

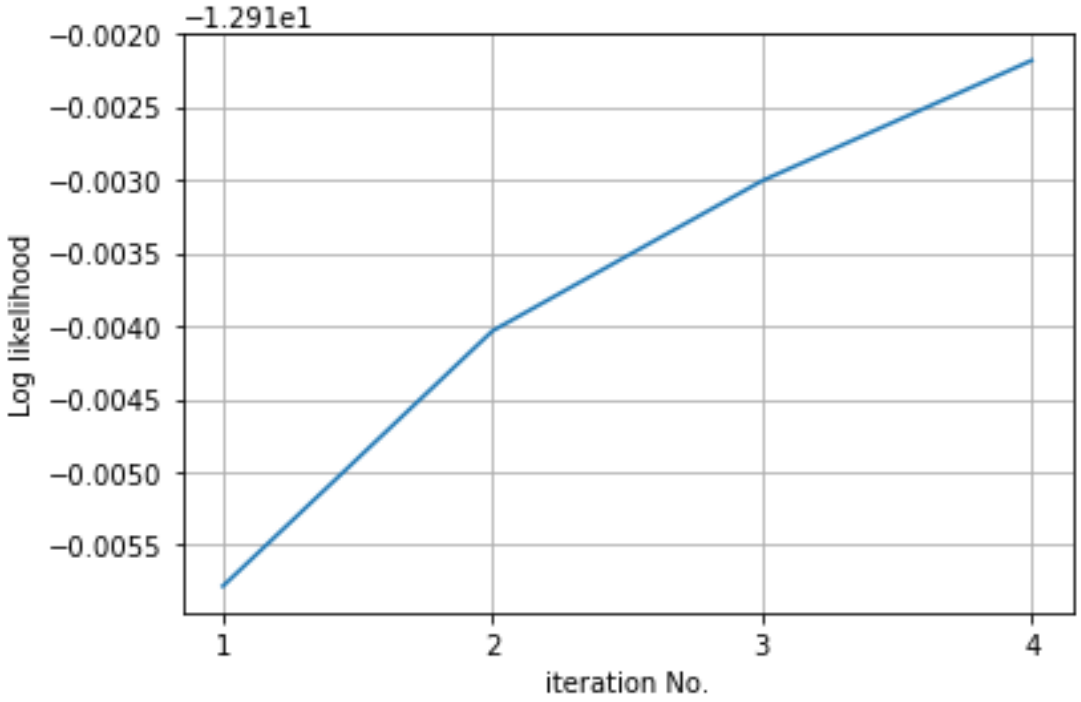
P(H=0/G=1): 0.1890502962045198

P(H=1/G=1): 0.8109497037954803

Log likelihood: -12.912184398191838

change value at the final iteration (diff value): 0.0008236665406347043

Iteration round: 4



5. hw2dataset_100

File name: hw2dataset_100.txt

Starting point of the parameters table.

P(G=0): 0.7
P(G=1): 0.30000000000000004
P(W=0/G=0): 0.8
P(W=1/G=0): 0.19999999999999996
P(W=0/G=1): 0.4
P(W=1/G=1): 0.6
P(H=0/G=0): 0.7
P(H=1/G=0): 0.30000000000000004
P(H=0/G=1): 0.3
P(H=1/G=1): 0.7

Parameters table at iteration No.1

P(G=0): 0.7254877848827677
P(G=1): 0.27451221511723223
P(W=0/G=0): 0.815006497542234
P(W=1/G=0): 0.18499350245776602
P(W=0/G=1): 0.3960579363894282
P(W=1/G=1): 0.6039420636105718
P(H=0/G=0): 0.7645262444205887
P(H=1/G=0): 0.2354737555794113
P(H=0/G=1): 0.3473271614155593
P(H=1/G=1): 0.6526728385844407

Log likelihood: -10.758755587490556
Parameters table at iteration No.2

P(G=0): 0.7239185720886191
P(G=1): 0.2760814279113809
P(W=0/G=0): 0.8204402934883277
P(W=1/G=0): 0.1795597065116723
P(W=0/G=1): 0.3841911245402892
P(W=1/G=1): 0.6158088754597109
P(H=0/G=0): 0.7701584326161355
P(H=1/G=0): 0.22984156738386452
P(H=0/G=1): 0.3349301975075245
P(H=1/G=1): 0.6650698024924755

Log likelihood: -10.754007642181621
Parameters table at iteration No.3

P(G=0): 0.7227915444553616
P(G=1): 0.2772084555446385
P(W=0/G=0): 0.824340849447788

P(W=1/G=0): 0.17565915055221204
P(W=0/G=1): 0.37579446869799293
P(W=1/G=1): 0.6242055313020071
P(H=0/G=0): 0.7741939998902739
P(H=1/G=0): 0.22580600010972607
P(H=0/G=1): 0.3261773632899694
P(H=1/G=1): 0.6738226367100306

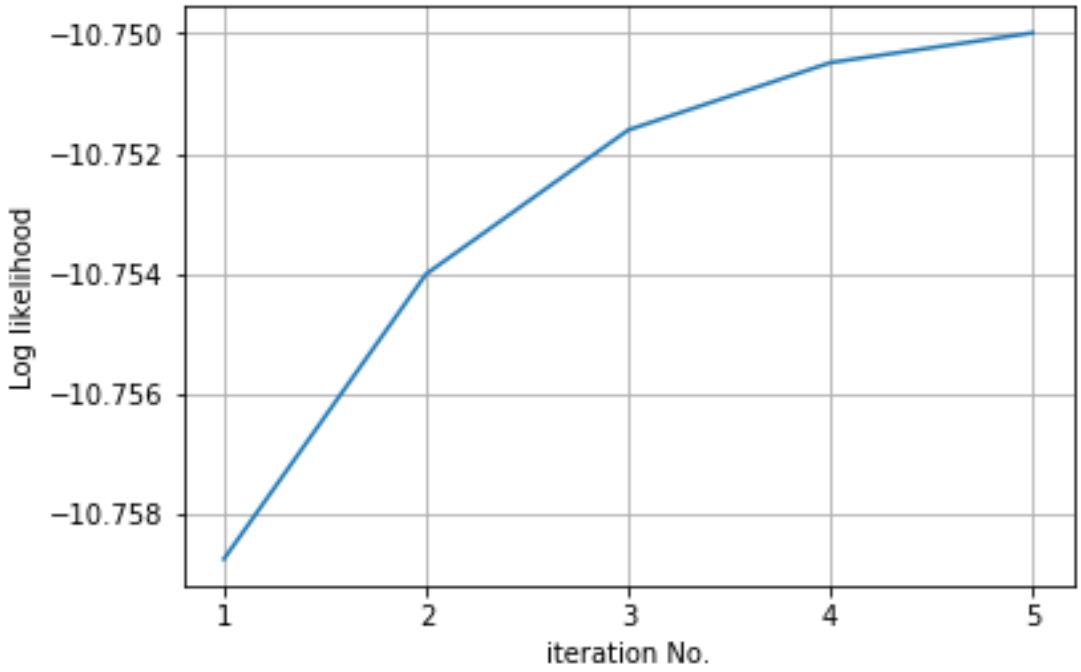
Log likelihood: -10.751607122534802
Parameters table at iteration No.4

P(G=0): 0.7220166006054971
P(G=1): 0.27798339939450256
P(W=0/G=0): 0.8270243645952018
P(W=1/G=0): 0.1729756354047982
P(W=0/G=1): 0.3700749033256316
P(W=1/G=1): 0.6299250966743684
P(H=0/G=0): 0.7769673189679055
P(H=1/G=0): 0.2230326810320945
P(H=0/G=1): 0.3202230700506577
P(H=1/G=1): 0.6797769299493424

Log likelihood: -10.750485009178107
Parameters table at iteration No.5

P(G=0): 0.7214998852443159
P(G=1): 0.2785001147556841
P(W=0/G=0): 0.8288151047172454
P(W=1/G=0): 0.1711848952827546
P(W=0/G=1): 0.36628350098613316
P(W=1/G=1): 0.6337164990138668
P(H=0/G=0): 0.7788167344820698
P(H=1/G=0): 0.22118326551793022
P(H=0/G=1): 0.3162792788150406
P(H=1/G=1): 0.6837207211849594

Log likelihood: -10.74998933220325
change value at the final iteration (diff value): 0.0004956769748574175
Iteration round: 5



Python Code

```
# -*- coding: utf-8 -*-
"""
Created on Sat Nov 11 21:11:56 2017

@author: topbu
"""

import numpy
import math
import matplotlib.pyplot as plt
from collections import defaultdict
import copy

class EM:
    def __init__(self, threshold):
        self.likeli = 0
        self.threshold = threshold
        self.data = []
        self.col_name = []
        self.count = 0 # count number of data lines
        self.p_gender = {'0':0, '1':0}
        self.p_weight = {c: defaultdict(int) for c in self.p_gender}
```

```
        self.p_height = {c: defaultdict(int) for c in self.p_gender}
        self.itera = 0 # count iterated round

    def parse(self, fileName):
        # get dataset
        with open(fileName, 'r', errors="replace") as text:
            tokens = text.readlines()

        # get the column name
        self.col_name = tokens[0].split()
        for i in range(1, len(tokens)):
            line_str = tokens[i].split()
            self.data.append(line_str)
            self.count += 1

        # set the missing data to be dictionry {'0':0, '1':0}
        for i in self.data:
            if i[0] is '-':
                # set dict for gender prob
                i[0] = {'0':0, '1':0}

        return

    def pickStarPot(self, init_v):
        # set the missing data to be dictionry {'0':1, '1':0}
        self.p_gender['0'] = init_v[0]
        self.p_gender['1'] = 1 - self.p_gender['0']

        self.p_weight['0']['0'] = init_v[1]
        self.p_weight['0']['1'] = 1 - self.p_weight['0']['0']
        self.p_weight['1']['0'] = init_v[2]
        self.p_weight['1']['1'] = 1 - self.p_weight['1']['0']

        self.p_height['0']['0'] = init_v[3]
        self.p_height['0']['1'] = 1 - self.p_height['0']['0']
        self.p_height['1']['0'] = init_v[4]
        self.p_height['1']['1'] = 1 - self.p_height['1']['0']

        print("\n")
        print("Starting point of the parameters table.")
        print("-----------------------------------")
        print("P(G=0): " + str(self.p_gender['0']))
        print("P(G=1): " + str(self.p_gender['1']))
```

```
        print("P(W=0/G=0): " + str(self.p_weight['0']['0']))
        print("P(W=1/G=0): " + str(self.p_weight['0']['1']))
        print("P(W=0/G=1): " + str(self.p_weight['1']['0']))
        print("P(W=1/G=1): " + str(self.p_weight['1']['1']))

        print("P(H=0/G=0): " + str(self.p_height['0']['0']))
        print("P(H=1/G=0): " + str(self.p_height['0']['1']))
        print("P(H=0/G=1): " + str(self.p_height['1']['0']))
        print("P(H=1/G=1): " + str(self.p_height['1']['1']))

        print("-------------------------------------")

        return

    def learn_params(self):
        # init all probability count
        c_gender = {c: defaultdict(int) for c in self.p_gender}
        c_gender['0'] = 0
        c_gender['1'] = 0

        c_weight = {c: defaultdict(int) for c in self.p_gender}
        c_weight['0']['0'] = 0
        c_weight['0']['1'] = 0
        c_weight['1']['0'] = 0
        c_weight['1']['1'] = 0

        c_height = {c: defaultdict(int) for c in self.p_gender}
        c_height['0']['0'] = 0
        c_height['0']['1'] = 0
        c_height['1']['0'] = 0
        c_height['1']['1'] = 0

        # iterate each dataset to count
        for l in self.data:

            if l[0] is '0': # male
                # count gender probability P(g=0)
                c_gender['0'] += 1

                # count weight probability P(w/g) -----p_weight[given_g=0][prob_w]
                c_weight['0'][l[1]] += 1

                # count height probability P(w/g) -----p_height[given_g=0][prob_h]
                c_height['0'][l[2]] += 1
```

```
        elif l[0] is '1': # female
            # count gender probability P(g=1)
            c_gender['1'] += 1

            # count weight probability P(w/g) -----p_weight[given_g=1][prob_w]
            c_weight['1'][l[1]] += 1

            # count height probability P(w/g) -----p_height[given_g=1][prob_h]
            c_height['1'][l[2]] += 1

        else: # estimate parameters using the complete data
            # count gender probability P(g=0)
            c_gender['0'] += l[0]['0']
            c_gender['1'] += l[0]['1']

            # count weight probability P(w/g) -----p_weight[given_g=0][prob_w]
            c_weight['0'][l[1]] += l[0]['0']
            c_weight['1'][l[1]] += l[0]['1']

            # count height probability P(w/g) -----p_height[given_g=0][prob_h]
            c_height['0'][l[2]] += l[0]['0']
            c_height['1'][l[2]] += l[0]['1']

    # calculate the parameters table
    # store the previous parameters table
    prev_p_gender = copy.deepcopy(self.p_gender)
    prev_p_weight = copy.deepcopy(self.p_weight)
    prev_p_height = copy.deepcopy(self.p_height)

    # calculate the new parameters table
    self.p_gender['0'] = c_gender['0'] / self.count
    self.p_gender['1'] = c_gender['1'] / self.count

    self.p_weight['0']['0'] = c_weight['0']['0'] / sum(c_weight['0'].values())
    self.p_weight['0']['1'] = 1 - self.p_weight['0']['0']
    self.p_weight['1']['0'] = c_weight['1']['0'] / sum(c_weight['1'].values())
    self.p_weight['1']['1'] = 1 - self.p_weight['1']['0']

    self.p_height['0']['0'] = c_height['0']['0'] / sum(c_height['0'].values())
    self.p_height['0']['1'] = 1 - self.p_height['0']['0']
    self.p_height['1']['0'] = c_height['1']['0'] / sum(c_height['1'].values())
    self.p_height['1']['1'] = 1 - self.p_height['1']['0']
```

```
        print("Parameters table at iteration No." + str(self.itera))
        print("-----------------------------------")
        print("P(G=0): " + str(self.p_gender['0']))
        print("P(G=1): " + str(self.p_gender['1']))

        print("P(W=0/G=0): " + str(self.p_weight['0']['0']))
        print("P(W=1/G=0): " + str(self.p_weight['0']['1']))
        print("P(W=0/G=1): " + str(self.p_weight['1']['0']))
        print("P(W=1/G=1): " + str(self.p_weight['1']['1']))

        print("P(H=0/G=0): " + str(self.p_height['0']['0']))
        print("P(H=1/G=0): " + str(self.p_height['0']['1']))
        print("P(H=0/G=1): " + str(self.p_height['1']['0']))
        print("P(H=1/G=1): " + str(self.p_height['1']['1']))

        print("-----------------------------------")
        return

    def estimate_missing_data(self):
        # estimate values of each missing dataset
        for i in self.data:
            if i[0] is not '0' and i[0] is not '1':
                # calc the likelihood for each gender in the current dataset
                i[0]['0'] = ((self.p_weight['0'][i[1]] * self.p_height['0'][i[2]] * self.p_gender['0'] )
                             / (self.p_gender['0'] * self.p_weight['0'][i[1]] * self.p_height['0'][i[2]] +
self.p_gender['1'] * self.p_weight['1'][i[1]] * self.p_height['1'][i[2]]))
                i[0]['1'] = 1 - i[0]['0']

        return

    def likeliHood(self):
        # store previous likelihood
        prev_likeli = self.likeli
        likeli = 1

        # calculate likelihood
        for i in self.data:
            if i[0] is '0':
                likeli *= self.p_gender['0']*self.p_weight['0'][i[1]]*self.p_height['0'][i[2]]
            elif i[0] is '1':
                likeli *= self.p_gender['1']*self.p_weight['1'][i[1]]*self.p_height['1'][i[2]]
            elif i[0] is not '0' and i[0] is not '1':
                likeli *= ((self.p_gender['0']*self.p_weight['0'][i[1]]*self.p_height['0'][i[2]])
                           + (self.p_gender['1']*self.p_weight['1'][i[1]]*self.p_height['1'][i[2]]))
```

```
            # calculate the log likelihood
            self.likeli = math.log10(likeli)

            # calculate the different
            return abs(self.likeli - prev_likeli)


def main():
    ####################
    # parse file
    ####################
     # create a NaiveBayesian Classification object with threshold = 0.001
    model = EM(0.001)

    # parse the training data
    fileName = "hw2dataset_100.txt"
    print("File name: " + str(fileName))
    model.parse(fileName)

    ######################################
    # Pick a starting point of the parameters
    ######################################

    #init vector =
    #              [P(G=0), P(W=0/G=0), P(W=0/G=1), P(H=0/G=0),    P(H=0/G=1)]
    init_v = [0.7,            0.8,                0.4,            0.7,                0.3]

    model.pickStarPot(init_v)

    # init diff to detect convergence
    diff = model.threshold + 1

    # record each iteration likelihood to plot
    li_record = []


    # Prodedure guaranteed to improve at each iteration, threshold = 0.001
    while diff > model.threshold:
        model.itera += 1
        ######################################
        # Complete the data using the current parameters
        ######################################
```

```
            model.estimate_missing_data()

            ########################################
            # Estimate the parameters related to data completion
            ########################################
            model.learn_params()

            # calculate log likelihood change
            diff = model.likeliHood()

            # record the log likelihood value
            li_record.append(model.likeli)
            print("Log likelihood:", model.likeli)


        print("change value at the final iteration (diff value): " + str(diff))


        print("Iteration round: " + str(model.itera))

        plt.plot(numpy.arange(1, model.itera+1, 1), li_record)
        plt.xlabel('iteration No.')
        plt.ylabel('Log likelihood')
        plt.xticks(numpy.arange(1, model.itera+1, 1))
        plt.grid()

if __name__ == '__main__':
        main()
```