Garvit Pugalia
504628127

COM SCI 145 – Homework #1

Linear Regression

(a) Report the learned weights and MSE (Mean Square Error) in the test dataset for each version, are they the same and why?

Closed-form solution

Weights: [-0.0862246 0.05340575 0.65803045 0.41731923 -0.01772481 0.30069864
1.02871152 0.48383363 0.26685697 0.04573456 0.31944742 1.14776959
0.29366213 0.41491543 0.85180482 -0.05950309 0.47235562 0.46198106
0.00497427 0.0205398 0.41310473 0.98508025 0.15573467 0.8618602
0.41974331 -0.06893699 0.33317496 0.27766637 -0.04184791 -0.23599504
0.15020297 0.37745027 0.80256455 0.16053288 0.2744667 0.63461071
0.74135259 0.56079776 0.94058723 -0.0432542 0.80803615 0.93967722
0.12225161 -0.19933624 0.09398732 0.11412993 0.35479619 0.78582876
0.38900433 0.11804526 0.67618837 0.70377377 0.05526258 -0.24919095
0.87339793 -0.01381723 0.83138416 0.90569236 0.39980648 0.25235308
0.69692397 -0.00949757 0.17676599 0.45822485 0.02743899 1.16718165
0.04176352 1.01993881 0.56015024 -0.29761224 0.3177761 0.55781578
1.1376088 0.55190283 0.4099807 0.91987238 1.34076835 0.53297825
0.63648277 0.22140583 0.21469531 -0.00609269 0.82898663 0.46891532
-0.25571565 0.1972989 1.38639797 0.87219453 0.65782257 0.54983464
1.11698567 0.94267463 0.79030138 0.30055848 0.53288973 0.22873689
0.86702876 0.98591924 0.08132528 0.30834368 0.70121488]

MSE: 4.396097860818858

Batch gradient descent

Weights: [0.53876388 0.05597506 0.7036856 0.44180875 0.5251787 0.38138321
0.70483297 0.47290786 0.2200791 0.2978031 0.51224097 0.72142551
0.72163757 0.75394006 0.58360988 0.10229932 0.3958301 0.42762901
0.42461638 0.57691052 0.4147374 0.92686179 0.16703347 0.65381537
0.31552114 0.45026224 0.36609707 0.25393948 0.27367766 0.21395953
0.19253094 0.52108356 0.43326551 0.46400735 0.56758441 0.22780981
0.496364 0.7381299 0.50976063 0.03056352 0.81072548 0.43037879
0.45844142 0.45021501 0.32576402 0.47106383 0.69621297 0.25398062
0.14245384 0.57155983 0.84129709 0.15462226 0.59803397 0.537751
0.36086354 0.07786574 0.37917239 0.35609094 0.63085625 0.64221498
0.58791609 0.35549635 0.02914768 0.75593386 0.20603376 0.44209351
0.4604963 0.4272972 0.33911404 0.11403783 0.09599335 0.721655
0.64668807 0.46306442 0.62257538 0.48768319 0.92989938 0.6806173
0.21208156 0.21397791 0.22756865 0.31336289 0.58473427 0.28505181
0.10623052 0.24460488 0.81544685 0.55528616 0.50926676 0.8208998

0.77255467 0.36288945 0.37880829 0.07204091 0.35605974 0.12467168
0.74322385 0.42757491 0.45093302 0.34637757 0.17154696]
MSE: 5.021485064190237

Stochastic gradient descent
Weights: [1.33236335e-01 -1.34589176e-03 7.04780783e-01 3.94749498e-01
9.66993607e-03 3.63583098e-01 1.08521145e+00 5.45591063e-01
3.21517562e-01 6.74076542e-02 3.20071432e-01 1.13478304e+00
2.91400226e-01 4.30081241e-01 9.18405906e-01 -2.16746329e-02
5.34313145e-01 4.56232273e-01 -1.12189485e-02 -1.48608405e-02
4.14953665e-01 9.90226750e-01 2.20711428e-01 8.92544304e-01
4.56560237e-01 -8.19275685e-02 3.26192044e-01 2.56609145e-01
2.90420360e-02 -2.32061114e-01 1.12781661e-01 3.70988640e-01
7.65022422e-01 7.98385777e-02 2.87720047e-01 5.98140094e-01
7.65282493e-01 5.62350532e-01 9.53828224e-01 -7.13207573e-02
7.90042148e-01 9.21463330e-01 1.44476916e-01 -2.02516844e-01
9.97324218e-02 8.34321327e-02 4.61488350e-01 8.39411606e-01
4.42878318e-01 6.78411437e-02 6.93732728e-01 7.50688783e-01
1.07977933e-01 -2.51936343e-01 8.32497525e-01 -3.08930097e-02
8.39941649e-01 9.33851270e-01 4.30514802e-01 3.41143545e-01
7.64383486e-01 3.23256363e-02 1.23664479e-01 5.52075760e-01
7.55495735e-02 1.26261348e+00 1.92651245e-02 1.09086054e+00
5.91286977e-01 -1.86913230e-01 3.26793250e-01 5.41552976e-01
1.16382997e+00 5.24958169e-01 3.98276768e-01 8.85255227e-01
1.31792372e+00 4.80094287e-01 6.64194626e-01 1.56126049e-01
2.10037894e-01 -1.90054221e-02 8.12143273e-01 3.82166866e-01
-3.24945176e-01 2.44297416e-01 1.39716507e+00 7.61023739e-01
7.81417826e-01 5.60520265e-01 1.04290136e+00 8.71722482e-01
8.62716006e-01 3.01612798e-01 5.17701439e-01 2.32040972e-01
8.65594545e-01 1.03818456e+00 5.65821511e-02 2.71216986e-01
6.00721457e-01]
MSE: 4.663885009866522

Are the reported weights and MSE the same for all versions, and why?

The reported weights and MSE are slightly different for different approaches. While a closed-form solution approach outputs the same weights and MSE for every trial, the other two approaches involve a randomness factor (i.e. the initialization of Beta) that can produce different weights and, consequently, MSEs for different runs. This randomness can cause the algorithms to find different minimas for different trials, thereby changing the weights. However, since each algorithm is effective with small datasets, the MSE (or Mean Square Error) on the test data is relatively similar for all versions.

Furthermore, the closed-form solution works well for smaller datasets with fewer complications, and hence is faster for this data. On the other hand, the batch gradient descent approach is the slowest, as it takes a step for each pass over the dataset i.e. requires many passes through the entire dataset.

(b) Apply z-score normalization for each feature x and report whether the normalization affect *B* and MSE (Mean Square Error) in the test dataset, for all three versions of the algorithm, and why?

Closed-form solution with normalization

Weights: [2.27729720e+01 1.53267685e-01 1.85400036e-01 1.20001101e-01
-5.02894960e-03 8.91855522e-02 2.85477509e-01 1.40249729e-01
7.58001703e-02 1.29653087e-02 9.40114997e-02 3.31501951e-01
8.48405150e-02 1.19998020e-01 2.42101087e-01 -1.70904428e-02
1.37119556e-01 1.35350218e-01 1.41619004e-03 5.96423043e-03
1.15830867e-01 2.84837752e-01 4.40248244e-02 2.49185633e-01
1.20285952e-01 -1.97966211e-02 9.78939759e-02 8.05403060e-02
-1.21241111e-02 -6.77059821e-02 4.42940642e-02 1.07814670e-01
2.27982170e-01 4.72154203e-02 7.98729034e-02 1.82957097e-01
2.10609705e-01 1.62079663e-01 2.74455584e-01 -1.24456123e-02
2.32346197e-01 2.68821067e-01 3.49745502e-02 -5.73174263e-02
2.74558199e-02 3.22366923e-02 1.03219840e-01 2.23792899e-01
1.12445398e-01 3.34223468e-02 1.96611852e-01 2.04171370e-01
1.61259528e-02 -7.12316220e-02 2.51757075e-01 -3.88735810e-03
2.31055679e-01 2.65481860e-01 1.14239087e-01 7.19519080e-02
2.03225977e-01 -2.77922653e-03 5.10043840e-02 1.31478537e-01
7.74623329e-03 3.36781203e-01 1.19518825e-02 2.98145298e-01
1.64253970e-01 -8.57326109e-02 9.04810592e-02 1.57878654e-01
3.30578812e-01 1.58142457e-01 1.17519641e-01 2.66603450e-01
3.90619100e-01 1.54573813e-01 1.82230684e-01 6.25165215e-02
6.11873098e-02 -1.74345404e-03 2.34361003e-01 1.35158424e-01
-7.34879378e-02 5.72871764e-02 4.02966409e-01 2.50642329e-01
1.87572968e-01 1.57855445e-01 3.18225717e-01 2.66144412e-01
2.29911686e-01 8.53225833e-02 1.56706806e-01 6.57087894e-02
2.52781623e-01 2.90068806e-01 2.33284776e-02 9.01229905e-02
2.03998476e-01]

MSE: 4.396097860818445

Batch gradient descent with normalization

Weights: [2.26254952e+01 1.63201521e-01 1.60477953e-01 1.25516954e-01
-1.12372278e-02 7.77021034e-02 3.09539786e-01 1.70146526e-01
8.71929489e-02 2.40614022e-02 1.05548711e-01 3.52222516e-01
8.38409711e-02 1.46445515e-01 2.43533026e-01 -5.80291610e-03
1.32946944e-01 1.39386552e-01 3.61768944e-03 1.80641033e-03
1.22844190e-01 2.89107061e-01 7.12834567e-02 2.61817897e-01
1.10896473e-01 -1.96395950e-02 1.14013897e-01 9.89535405e-02

2.54333437e-02 -3.47217183e-02 3.84135979e-02 1.33619497e-01
2.46569579e-01 3.54196691e-02 1.01967313e-01 1.86031185e-01
2.00662275e-01 1.70074662e-01 3.00873892e-01 -9.17023831e-03
2.49405064e-01 2.70257662e-01 3.52423393e-02 -3.44583124e-02
3.61555932e-02 3.56042903e-02 9.89019413e-02 2.15785047e-01
1.06282453e-01 3.77713540e-02 1.97664815e-01 2.10495986e-01
3.77715826e-02 -6.75068306e-02 2.52644075e-01 5.61826960e-03
2.38806129e-01 2.77297584e-01 1.16345684e-01 7.19183871e-02
2.05952074e-01 7.60362262e-03 3.77469905e-02 1.40306680e-01
3.17897619e-02 3.51795550e-01 1.97233402e-02 3.18306935e-01
1.80930670e-01 -6.96451409e-02 9.43925222e-02 1.74237062e-01
3.57967290e-01 1.82074785e-01 1.05922476e-01 2.68899636e-01
4.10942803e-01 1.60986624e-01 1.77368920e-01 5.35632818e-02
6.62740248e-02 3.31242602e-03 2.38275117e-01 1.54228669e-01
-6.50735720e-02 7.26901537e-02 3.87857533e-01 2.28869643e-01
1.80440146e-01 1.59829227e-01 3.27807291e-01 2.59230676e-01
2.45942764e-01 9.44077141e-02 1.52676660e-01 8.08122487e-02
2.73090158e-01 2.78820193e-01 3.85578603e-02 9.08819534e-02
1.88450342e-01]
MSE: 4.410062880158726

Stochastic gradient descent with normalization
Weights : [2.27395517e+01 1.77760445e-01 5.46806101e-02 1.33097891e-01
2.43116316e-02 1.30414208e-01 4.35748651e-01 1.97806378e-01
-3.07705329e-02 8.92773478e-02 1.50845099e-01 2.84592278e-01
9.22016879e-02 1.63408247e-01 2.73364836e-01 1.89900431e-02
3.65301963e-02 7.37175773e-03 8.00250191e-02 7.68929459e-02
1.12185552e-01 2.79432286e-01 -9.75375590e-02 8.10493234e-02
2.20237847e-01 6.88606890e-02 1.04534318e-01 2.04192976e-01
1.09517252e-02 -9.78947463e-02 1.13778927e-01 6.78535766e-02
1.98351524e-01 2.32777207e-01 -3.36284333e-02 8.40528136e-02
1.46960616e-01 1.24587513e-01 4.59636550e-01 -1.57918457e-01
2.39203000e-01 3.73520720e-01 1.49530219e-01 -5.09549154e-02
2.03976759e-01 1.63405023e-01 5.73600825e-02 1.16981962e-01
6.92748708e-02 3.00204687e-02 2.10997794e-01 3.48279918e-01
1.54657219e-01 -2.71191271e-01 2.44144836e-01 -7.89576281e-02
1.79000698e-01 1.66426995e-01 2.64532675e-01 -5.32874877e-02
1.52333594e-01 -1.87281532e-01 1.83489527e-01 1.29195196e-01
9.18492352e-02 3.08946777e-01 9.68661293e-02 3.37971620e-01
2.21042814e-01 -1.29791690e-01 1.73605861e-01 4.48798462e-02
4.09540422e-01 1.35240790e-01 2.20090638e-01 2.67242389e-01
4.74023600e-01 1.39662455e-01 8.27387951e-02 8.68925723e-02
8.97726317e-03 7.56585728e-02 2.69175196e-01 2.24139589e-01
-2.00810699e-01 6.67895367e-02 3.66410429e-01 3.64197240e-01

```
        3.89607109e-01  1.88236526e-01  3.86614628e-01  3.14397897e-01
        3.06692774e-01  3.37772132e-02  2.37857719e-01  1.01846382e-01
        4.29519165e-01  3.95512216e-01 -2.86737805e-02  2.13248481e-02
        1.76966723e-01]
MSE:  5.388017322313453
```

How does the normalization affect weights and MSE for all versions, and why?

The normalization affects the weights for all three versions. This is because normalization is the act of adjusting the values to fit a certain scale. In this case, all the values of weight have been brought closer to zero, therefore, they are specifically very small positive or negative numbers, as compared to values seen without normalization.

The MSE also changes for all three versions. However, this change is minimal in the closed-form solution approach (of order of magnitude = 10^{-12}). The change in MSE is much more significant in the gradient descent approaches. For example, in the stochastic gradient descent approach, the MSE for normalized trials are consistently higher than the MSE for unnormalized trials. This is possibly because normalization over-corrects for the different range of feature values, which could indicate that the data has a high range of values.

Garvit Pugalia
504628127

Logistic Regression and Model Selection

(a) Report the learned weights and accuracy (Mean Square Error) in the test dataset for each version, are they the same and why? Discuss the pros and cons of the two methods.

Batch gradient descent
Weights: [8.67142427 -9.20743776 -5.01444896 -6.05227229 -0.70983265]
Training avgLogL: 0.023490256075182334
Test accuracy: 0.9890510948905109

Newton Raphson method
Weights: [1.17361791 -0.96352927 -0.53124574 -0.67260408 -0.01317662]
Training avgLogL: 0.10347000230760538
Test accuracy: 0.9927007299270073

Are the learned weights and accuracy in the test dataset the same for each version, and why?

While the learned weights do vary between versions and (slightly) between runs, they are oddly consistent between algorithms. For example, looking at the reported weights above, while the values for Newton-Raphson are lower, the relative magnitudes between the values are very similar. (If the weights were sorted for both algorithms, the positions of the weight will be in the same order)

The accuracy for the batch gradient descent algorithm stays at ~98.91% consistently, whereas the accuracy for the Newton-Raphson method varies between 96-99%. In layman terms, the batch gradient descent approach incrementally steps towards the zero of the function, whereas Newton-Raphson attempts to calculate and directly reach (*what it believes to be*) the zero of the function, taking more time in computation. This can lead to a variation in accuracy based on the estimations by the Newton-Raphson approach, and can therefore be better or worse than the accuracy of a simple, batch gradient descent approach.

(b) Discuss how regularization is affecting the training loss (in terms of average log loss) and test accuracy based on the experimental results.

Regularization has a counterintuitive effect on the training loss and test accuracy. However, this is also dependent on the value of lambda (in the code, variable ‘reg’). For example, with a regularization parameter lambda = 0.005, we get the following values:

Weights: [1.37461606 -1.48961326 -0.84903538 -0.94848147 -0.17862414]
Training avgLogL: 0.05995467030630324
Test accuracy: 0.9890510948905109

And with a regularization parameter lambda = 0.05, we get very different values:

Weights: [0.48311729 -0.8600081 -0.46079049 -0.45532274 -0.16806671]
Training avgLogL: 0.13608486919680954
Test accuracy: 0.9671532846715328

Firstly, as with normalization in linear regression, the regularization leads to smaller magnitude of weights. However, the training loss and test accuracy are dependent on the value of lambda. With a small value of lambda, the algorithm performs similar to the batch gradient descent (possibly due to lack of restrictions). However, surprisingly, as the value of lambda and, consequently, the restrictions increase, the algorithm begins to perform less accurately. This can be attributed to the fact that the non-regularized algorithm already performed at 98% accuracy. The data, therefore, might already be fitting properly into the algorithm, and any form of regularization leads to underfitting and worsening accuracy. The average log loss follows a similar pattern, increasing with lambda, due to the bias introduced by regularization.