Homework 1

Rob Harries
904836501

1. Linear Regression

1.1.

Before Normalization:

Method 0-0: closed-form, not normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [5.17285600e-01 -1.70173005e-02 -1.25229040e-02 -2.37364105e-02
-7.26850224e-03 -1.75015833e-03 -2.54105104e-02 -2.95826147e-02
-1.54205779e-02 -8.11041550e-03 -7.73115897e-03 2.23326880e-02
1.08377404e-02 -1.78034252e-02 -1.42799326e-02 1.10863872e-03
5.58962376e-03 -1.66481000e-02 1.74175345e-02 -9.79321083e-03
-1.13160627e-02 -2.43621012e-02 1.22042094e-02 -2.22008980e-02
-8.08868427e-03 1.98280275e-02 -1.62549170e-02 1.57163255e-02
5.55093555e-03 2.52723067e-02 -1.79696813e-02 -3.42412589e-02
2.33967228e-02 -1.18951150e-02 -8.29832518e-03 1.08683008e-03
1.07503176e-02 5.89595929e-03 -1.42884432e-02 -7.60366278e-03
-3.59068468e-03 -2.43039502e-02 -1.50352102e-02 -4.91648480e-05
-1.75975159e-02 -5.12186137e-03 -6.03505757e-03 2.11963730e-03
1.84672144e-02 5.97564034e-03 7.70482473e-03 -1.32971032e-02
-1.56211468e-02 1.64262479e-02 -1.87298040e-02 -2.62080745e-02
1.98841713e-02 -2.47382511e-02 7.11668306e-03 -2.56090472e-02
-1.43803106e-02 -1.78350545e-02 -2.34158378e-02 -1.21549137e-02
2.26194590e-02 -1.35242391e-02 8.88066425e-04 -1.42204055e-02
2.99114634e-03 5.22524532e-03 -1.79063948e-02 3.83684473e-03
8.33356729e-03 2.56888081e-02 -1.80756710e-02 -1.99695440e-02
-2.86138337e-02 2.35867028e-02 1.90433998e-03 1.72159943e-02
3.03296234e-02 1.74398815e-02 -2.78753941e-02 1.30140929e-02
2.60430914e-02 -2.59504768e-04 1.74699574e-02 3.43722771e-05
1.37552942e-02 2.24646356e-02 -1.22617221e-02 -1.82281224e-02
1.80041301e-02 -7.43819418e-04 -2.84486814e-02 -1.42173525e-02
-9.10220722e-04 -2.59410878e-02 1.86651575e-02 2.90379883e-02
-1.63292879e-03]

Training MSE: 0.086938866754
Test MSE: 0.110175402817

Method 1-0: batch descent, not normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [5.18915256e-01 -1.69937806e-02 -1.24751243e-02 -2.37437437e-02
-7.27478253e-03 -1.76392882e-03 -2.54168548e-02 -2.95716439e-02
-1.54003925e-02 -8.09240863e-03 -7.73438001e-03 2.23490094e-02
1.08501568e-02 -1.77637832e-02 -1.42298945e-02 1.12338555e-03
5.60234593e-03 -1.66825667e-02 1.74356772e-02 -9.78577167e-03
-1.13329282e-02 -2.43481818e-02 1.22240311e-02 -2.21708476e-02
-8.10459112e-03 1.98279635e-02 -1.62637600e-02 1.57047118e-02
5.54953968e-03 2.52510516e-02 -1.79744409e-02 -3.42427942e-02
2.33692270e-02 -1.19055424e-02 -8.31262204e-03 1.07611187e-03
1.07529103e-02 5.85240760e-03 -1.42723346e-02 -7.65480280e-03
-3.58728699e-03 -2.42941351e-02 -1.50105526e-02 -4.73988572e-05
-1.75969639e-02 -5.11632808e-03 -6.06413465e-03 2.07018493e-03
1.84498910e-02 5.97220882e-03 7.72807460e-03 -1.32902196e-02
-1.56096833e-02 1.64358714e-02 -1.87257209e-02 -2.62145772e-02
1.98328034e-02 -2.47373760e-02 7.16993767e-03 -2.56523258e-02
-1.43771805e-02 -1.78198091e-02 -2.34143636e-02 -1.21665757e-02
2.26231505e-02 -1.35466386e-02 9.14912848e-04 -1.42571943e-02
2.98616188e-03 5.18527838e-03 -1.79595158e-02 3.84152949e-03
8.29339130e-03 2.56789237e-02 -1.81048422e-02 -1.99823453e-02
-2.86171008e-02 2.35594379e-02 1.92869306e-03 1.72032904e-02
3.03121966e-02 1.74525879e-02 -2.78827423e-02 1.30245408e-02
2.60187842e-02 -2.89485634e-04 1.74845335e-02 1.16106538e-05
1.37487152e-02 2.24502734e-02 -1.22899099e-02 -1.82740044e-02
1.80184923e-02 -7.51529609e-04 -2.84555813e-02 -1.42457851e-02
-9.33713946e-04 -2.59308040e-02 1.86651691e-02 2.90312359e-02
-1.61723846e-03]

Training MSE: 0.0869413406236
Test MSE: 0.110229446339

Method 2-0: stochastic descent, not normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [5.17942917e-01 -1.67136251e-02 -7.96417959e-03 -2.92878975e-02
-1.18059912e-02 -3.54735600e-03 -2.05162117e-02 -2.83184809e-02

-1.31855074e-02 -9.06572094e-03 -1.14733924e-02 1.78571728e-02
1.13127424e-02 -1.91501137e-02 -1.08965601e-02 -8.10710857e-05
9.16375404e-03 -1.48190720e-02 1.39845777e-02 -4.75888856e-03
-1.02262984e-02 -2.07168997e-02 9.35924595e-03 -1.46650374e-02
-7.70039643e-03 1.26534509e-02 -1.35938709e-02 2.29198916e-02
7.80663719e-03 3.08125290e-02 -1.36169445e-02 -3.30786233e-02
2.14676363e-02 -1.14208018e-02 -2.40082594e-03 7.79035496e-04
1.40372596e-02 4.29074366e-03 -1.21737843e-02 -5.03752925e-03
-8.35072128e-04 -2.60844668e-02 -1.15421925e-02 -3.75683737e-03
-2.24287838e-02 -6.38411314e-03 -5.64515149e-03 3.62598860e-03
1.39625029e-02 1.01397216e-02 5.58129847e-03 -1.67919894e-02
-1.53271838e-02 1.73691827e-02 -1.65168855e-02 -2.46966099e-02
1.75609950e-02 -2.60449744e-02 3.87199339e-03 -2.31005022e-02
-1.39754357e-02 -2.01755683e-02 -2.13928792e-02 -1.01778180e-02
2.87326820e-02 -2.02471885e-02 1.20643998e-03 -1.39365592e-02
1.48717851e-03 5.64946073e-03 -2.08200566e-02 6.04126241e-03
5.99624127e-03 3.01348258e-02 -2.28920863e-02 -1.44223476e-02
-2.69019820e-02 2.53169411e-02 2.31752097e-03 1.69963419e-02
3.15078615e-02 1.68505138e-02 -2.50271869e-02 1.32270840e-02
2.41127953e-02 -1.18560251e-03 1.52608118e-02 -1.97420923e-03
1.56033162e-02 2.26599919e-02 -9.68540985e-03 -1.78995052e-02
1.93913780e-02 1.22366257e-03 -2.66038440e-02 -1.50282060e-02
-3.05573372e-04 -1.98801974e-02 2.08037529e-02 2.81669132e-02
-7.02565367e-04]

Training MSE: 0.0923832828267
Test MSE: 0.115179723455

Pros/Cons:

Each method produces different beta values and different test MSE's. Batch gradient descent has very similar beta values to the closed-form solution, with only slight variations due to the inherent noise in the initial setup. The closed-form solution has a slightly lower MSE for both test and training examples, which is expected, since it is the mathematically exact representation of the answer.

With Normalization

Method 0-1: closed-form, normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [5.23000000e-01 -3.95099505e-02 -3.01401932e-02 -5.71438644e-02
-1.72769796e-02 -4.13700127e-03 -5.86318630e-02 -6.89027284e-02
-3.56331805e-02 -1.87845537e-02 -1.82888714e-02 5.29276130e-02
2.53519018e-02 -4.15812928e-02 -3.30193382e-02 2.65867992e-03
1.34068950e-02 -3.88013327e-02 4.11038867e-02 -2.32239983e-02
-2.68494719e-02 -5.67582270e-02 2.85948574e-02 -5.22058491e-02
-1.94232592e-02 4.61988692e-02 -3.87491283e-02 3.82055256e-02
1.27021593e-02 5.82271850e-02 -4.20937718e-02 -8.05582038e-02
5.50688227e-02 -2.88202457e-02 -1.94706479e-02 2.58596756e-03
2.55048685e-02 1.39991237e-02 -3.38312079e-02 -1.80218433e-02
-8.42135902e-03 -5.61252496e-02 -3.60939866e-02 -1.12787490e-04
-4.02969672e-02 -1.20851201e-02 -1.41809480e-02 5.11770552e-03
4.48842190e-02 1.42864924e-02 1.79066117e-02 -3.08841654e-02
-3.67139837e-02 3.83560781e-02 -4.47435146e-02 -6.08180754e-02
4.69774181e-02 -5.86346690e-02 1.62361334e-02 -6.06942237e-02
-3.38205570e-02 -4.24317897e-02 -5.46648364e-02 -2.89378305e-02
5.33687506e-02 -3.17462303e-02 2.12826319e-03 -3.26837546e-02
6.84819052e-03 1.25455103e-02 -4.09640271e-02 8.88512549e-03
1.94883628e-02 6.04797247e-02 -4.23185183e-02 -4.76582979e-02
-6.69833777e-02 5.66019062e-02 4.63178581e-03 4.13664903e-02
7.10828556e-02 4.08986579e-02 -6.46605942e-02 3.05062530e-02
6.11970818e-02 -6.13118531e-04 4.12093831e-02 8.04511196e-05
3.21203863e-02 5.30651849e-02 -2.83935172e-02 -4.22856651e-02
4.23271015e-02 -1.72635991e-03 -6.75124152e-02 -3.30151234e-02
-2.14687553e-03 -6.00152621e-02 4.30059659e-02 6.79904935e-02
-3.84367853e-03]

Training MSE: 0.086938866754
Test MSE: 0.110175402817

Method 1-1: batch descent, normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [0.5213266 -0.03126679 -0.02222113 -0.03601736 -0.00722427 -0.00422804
-0.05341362 -0.07067559 -0.01502639 0.01170233 -0.00267908 0.04924776
0.03229433 -0.05539572 -0.01292373 0.00323187 0.03097273 -0.02690102
0.05978286 -0.01350912 -0.01732039 -0.01896992 0.04635564 -0.04502301
-0.01578912 0.04819893 -0.04247976 0.05835856 0.00680509 0.05516194
-0.01003991 -0.06361428 0.04744246 -0.01438821 -0.02290755 -0.01895538

0.0497496 0.04639241 -0.03534919 0.00515779 -0.01607726 -0.06524733
-0.02280139 -0.00197806 -0.01762703 -0.00932635 -0.00604674 -0.00096069
 0.06321714 0.02903768 0.01266522 -0.03026256 -0.0438365 0.03687244
-0.04498432 -0.05910256 0.05100284 -0.05425886 0.01552802 -0.06522846
-0.03193898 -0.04989939 -0.04806433 -0.0368185 0.06424144 -0.02533431
-0.00679912 -0.02292143 0.00378449 0.0381067 -0.03405998 0.01080083
 0.0140278 0.06534851 -0.03628541 -0.02482198 -0.0428101 0.06605098
 0.00822781 0.03919259 0.09943183 0.04864398 -0.05522695 0.0307136
 0.07235215 0.00075243 0.04242725 0.01048283 0.04825046 0.055005
-0.01242293 -0.0273741 0.04958834 0.01452668 -0.07804491 -0.01490766
 0.0023915 -0.03604111 0.02186797 0.07111968 0.00925175]

Training MSE: 0.0982517482867
Test MSE: 0.134767378089

Method 2-1: stochastic descent, normalized

Learning Rate: 0.005, Iterations = 10000

Beta: [0.53277346 -0.0563563 -0.02943188 -0.04487914 -0.04477916 0.00766831
-0.04121991 -0.05853206 -0.00457016 -0.02648927 -0.01484024 0.04450653
 0.05111861 -0.0413319 -0.02168384 0.00781379 0.01400191 -0.02346845
 0.02764945 -0.02662325 -0.01858523 -0.05129951 0.03070017 -0.04437313
-0.03466882 0.06144976 -0.02398974 0.05228517 -0.01744078 0.04077948
-0.0457092 -0.07254838 0.04497546 -0.04924585 -0.03061568 0.01202501
 0.02342062 0.00710746 -0.01097828 -0.01557824 -0.00784255 -0.03782551
-0.03538486 -0.0176389 -0.05630595 -0.01415391 -0.01257429 0.00586946
 0.03713687 0.03316254 0.02451777 -0.03721354 -0.03621582 0.04353579
-0.04074121 -0.05038535 0.03973803 -0.04971069 0.02513268 -0.06615219
-0.05154685 -0.02976791 -0.06887244 -0.02286867 0.07811162 -0.04724967
-0.01706756 -0.02941321 -0.00516432 0.01108615 -0.05260795 0.00586882
 0.01921841 0.07678571 -0.04685699 -0.03644587 -0.09732702 0.03177613
-0.00898472 0.02450143 0.06082645 0.03415317 -0.04576505 0.02568903
 0.09145752 -0.02369297 0.02516841 -0.01490692 0.02743699 0.08025729
-0.03867661 -0.04309527 0.06229694 0.00307423 -0.08536402 -0.03524359
 0.0129749 -0.07034076 0.02482941 0.06113227 -0.0316488]

Training MSE: 0.109048681503
Test MSE: 0.134940393571

Pros/Cons:

With normalization, beta values should change, as they are technically being applied to entirely different variables. This is the case for all three training approaches. In closed-form, the beta values are different, but the training and test MSE's are exactly the same, as expected of the mathematically maximized answer. For both batch descent and stochastic descent, performance decreased with normalization, which leads me to believe that some of the most important features for prediction were scaled larger than others, making them by default more impactful in prediction, making training easier. With normalization, this advantage disappeared.

1.2. Derive closed form with l2 regularization term

$$J = \frac{1}{2n}(X^T\beta - y)^T(X^T\beta - y) + \frac{\lambda}{2n}\beta^T\beta$$
$$J = \frac{1}{2n}(\beta^T X^T X\beta - y^T X\beta - \beta^T X^T y + y^T y + \lambda\beta^T\beta)$$
$$\frac{\partial J}{\partial\beta} = \frac{1}{2n}(2X^T X\beta - 2X^T y + 2\lambda\beta) = 0$$
$$\frac{\partial J}{\partial\beta} = X^T X\beta + \lambda\beta - X^T y = 0$$
$$(X^T X + \lambda I)\beta = X^T y$$
$$\beta = (X^T X + \lambda I)^{-1} X^T y$$

2. Logistic Regression

2.1.

Before Normalization

Method 0-0: batch descent, not normalized

Learning Rate: 0.000001, Iterations: 20000

Beta: [6.21291071e-01 4.11829124e-04 3.56249095e-02 4.60999079e-01 -3.42645467e-02 6.04786795e-01]

Training avgLogL: -0.56182164719
Test accuracy: 0.604373757455

Method 0-1: Newton-Raphson, not normalized

At first, the initial weights were too high, causing $x_i^T\beta$ to consistently be around ±1e30, meaning that the sigmoid function always returned either 0 or 1, and training never worked. In order to get around this, I initialized all beta values to 0, rather than rand().

Learning Rate = 0.0001, Iterations = 5000

Beta: [-3.22798095e+00 6.83377915e-04 2.08639870e-01
1.41271132e+01
-4.25052554e-02 3.16303300e+01]

Training avgLogL: -0.460100375351
Test accuracy: 0.753479125249

After Normalization

Method 0-1: batch descent, normalized

Learning Rate: 0.001, Iterations = 20000

Beta: [0.20863409 1.51806167 0.67140578 0.89822273
-0.49657551 0.51555082]

Training avgLogL: -0.471026351102
Test accuracy: 0.735586481113

Method 1-1: Newton-Raphson, normalized

Normalization eliminates the errors with extreme values found in the Newton-Raphson case without normalization, but ends up with the same accuracy, so probably has equivalent weights (accounting for the normalization).

Learning Rate = 0.0001, Iterations = 5000:

Beta: [0.27910793 2.15368204 1.23434665 1.32102078
-0.66170494 0.41580785]

Training avgLogL: -0.460100375351
Test accuracy: 0.753479125249

Pros/Cons

The Newton-Raphson method is slower than batch gradient descent, but seems to converge earlier and more reliably to a cost minimum. Newton-Raphson has the downside that with un-normalized features that have very large values, a randomly initialized set of weights can stop the method from learning correctly. This is easily avoided by initializing the weights to 0. When normalized, both of the methods converge earlier and more reliably to a minimum, since all features are prioritized equally by the learner.

2.2. Derive log likelihood derivative with l2 regularization

$$J(\beta) = -\sum_{i=1}^{n} \frac{y_i x_i^T \beta - log(1+e^{x_i^T \beta})}{n} + \lambda \sum_{j=1}^{p} \beta_j$$

$$\frac{\partial J}{\partial \beta_k} = -\sum_{i=1}^{n} \left(\frac{y_i x_{ik}}{n} - \frac{x_{ik} e^{x_i^T \beta}}{n(1+e^{x_i^T \beta})}\right) + 2\lambda\beta_k$$

$$\frac{\partial J}{\partial \beta_k} = -\sum_{i=1}^{n} \frac{x_{ik}}{n}\left(y_i - \frac{1}{1+e^{-x_i^T \beta}}\right) + 2\lambda\beta_k$$

$$\frac{\partial J}{\partial \beta_k} = -\sum_{i=1}^{n} \frac{x_{ik}}{n}(y_i - \sigma(x_i^T \beta)) + 2\lambda\beta_k$$

$$\frac{\partial J}{\partial \bar{\beta}} = -\frac{1}{n}\bar{x}^T(\bar{y} - \sigma(\bar{x}\,\bar{\beta})) + 2\lambda\bar{\beta}$$

2.3. Test batch gradient descent with l2 regularization

The effects of l2 regularization depend on the regularization parameter λ. At a value of 0.01, test accuracy decreases to ~0.46 and training loss increases dramatically. As λ increases, test accuracy and training loss both decrease. As λ decreases, test accuracy increases back up to what it was without regularization, which makes sense since the regularization term is getting smaller and having less of an effect on the derivative.