Optimal and Learning Control for Robotics Assignment 1

Seoho Kang

February 21, 2019

1 Convex optimization with linear equalities

* I follow the given rule of the question for Q, A, b

1.1 Write Lagrangian of the optimization problem and KKT conditions for optimality

$$L = \frac{1}{2}x^T Q x + \lambda^T (Ax - b)$$
$$KKTCondition1 : Ax = b$$
$$KKTCondition2 : \frac{\partial L}{\partial x} = 0 = \frac{1}{2}(Q + Q^T)x + A^T\lambda$$

1.2 Solve KKT System and find optimal lagrange multipliers as a function of Q, A, b

$\begin{pmatrix} \frac{1}{2}(Q+Q^T) & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} 0^T \\ b \end{pmatrix}$

* We can find x and lambda numerical way or we can inverse $\begin{pmatrix} \frac{1}{2}(Q+Q^T) & A^T \\ A & 0 \end{pmatrix}$ this matrix to derive x and lambda w.r.t. Q,A,b

1.3 Use above result to compute the minimum of the given function under the constraint that the sum of $x \in R^3$ should be equal to 1. What is the value of x and Lagrange Multiplier? Verify the constraint is satisfied for your result.

1) Under the constraint Ax = b, we can find $\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \mathrm{x}_1 + x_2 + x_3 = 1$

2) Using, the 1.2 results, we can compute $\begin{pmatrix} 10 & 1 & 0 & 1 \\ 1 & 100 & 2 & 1 \\ 0 & 2 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

We get $\mathrm{x}_1, x_2, x_3 : [[0.09090909][-0.01030928][0.91940019]]$

lagrange$_m ultiplier(\lambda) : [-0.89878163]$

3) Since, A = [1 1 1] and it is eliminated by gauss elimination to [1 0 0] which has rank 1. So it is full rank and the given constraint is satisfied.

2 Newton's Method

2.1 Find minimum point with the given 1D, 2D functions

1) I call first given function as 1D function and it's minimum was 1.0000000321720988 2) I call second given function as 2D function and it's minimum was 0.0 3) In order to tackle this problem I solved Hy = g (where H is Hessian and g is gradient) 4) Stopping function and every specific algorithm is implemented from the class material and please check my jupyter notebook

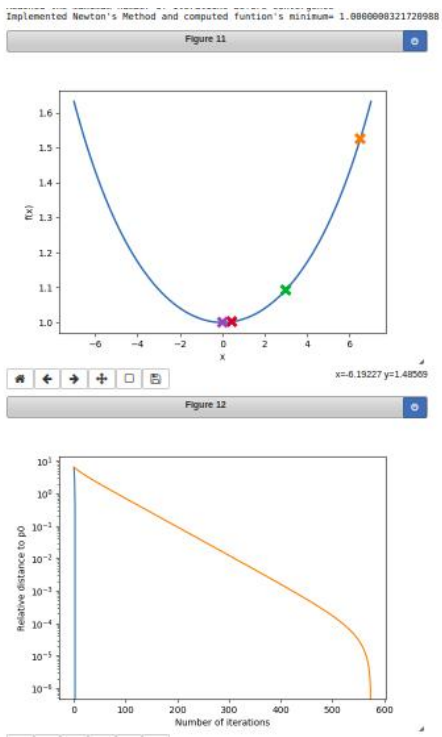

Figure 1: Upper graph shows how minimum of the function was found by Newton's Method which is close to 1. Lower graph compares Newton's Method(blue) with Gradient Descent(orange) and it indicates that Newton's method is much more efficient.

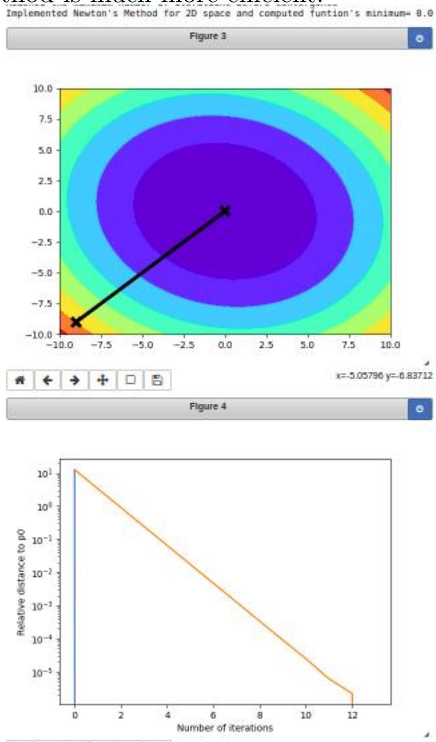

Figure 2: Upper graph shows how minimum of the function was found by Newton's Method in 2D space which is close to 0.0. Lower graph compares Newton's Method(blue) with Gradient Descent(orange) and it indicates that Newton's method is much more efficient

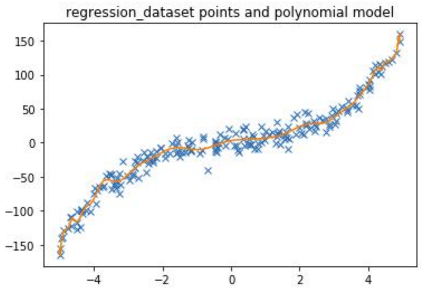

Figure 3: Fit looks good. It is much more sophisticated then low degree such as 3. We can tell the line follows data points very accurately.

3 Linear Least Squares

3.1 Best Polynomial fit for regression dataset

– What is the order of the polynomial that best fits the data? Why?
It is 157 and it has the smallest fitting error which is around 17383.4. I tackled this problem by finding solvable degree and it was 221. From degree 222, it outputs nan and I guess the reason as not enough various data points for degree 222. I looped through 0 to 221 and found the smallest fitting error where the degree was 157.
– What are the polynomial coefficient?
It starts from degree 0 to 157 : [3.24082778e+000 8.05773618e+000 -1.61123468e+001 8.24493502e+000 1.71931081e+001 -1.37194399e+001 -7.86087357e+000 7.63513027e+000 2.19558650e+000 -2.01001970e+000 -4.18255216e-001 2.79605772e-001 5.32287929e-002 -1.98043990e-002 -4.11723389e-003 4.59473132e-004 1.46664746e-004 2.07221699e-005 1.84958111e-006 -8.84784523e-007 -2.54371077e-007 -3.77100603e-008 -4.19878315e-009 2.31386474e-009 6.83078511e-010 -5.04504399e-011 -1.61610946e-011 2.95021162e-012 1.04927294e-012 -7.25310866e-014 -9.13982735e-014 -4.04721165e-015 8.57180848e-016 8.51085053e-017 1.42880825e-016 5.40785722e-018 -1.78857012e-018 1.23250097e-019 -7.04939583e-020 1.70630579e-021 -4.75546548e-021 -9.51134298e-022 1.76977001e-022 -1.77821406e-023 -1.18957272e-024 2.23895852e-024 -4.49454588e-026 -4.12038939e-027 5.91208846e-027 -1.03390972e-027 -8.16768264e-029 -1.22298272e-029 -4.20359236e-031 -2.56413924e-030 -3.57753545e-033 -2.63267173e-032 1.19514646e-032 6.43736161e-033 9.07958801e-034 4.39227802e-034 -6.26166872e-035 -7.93855732e-036 1.79402878e-036 -5.28948009e-037 3.14856563e-038 4.80876524e-039 -5.46671406e-039 6.82041549e-040 5.94247100e-041 -8.36283373e-042 -1.56697956e-042 -1.23541747e-042 -1.45585144e-043 -6.07547380e-044 -3.42058433e-045 1.61670592e-045 2.35030111e-046 3.44872272e-047 1.95522432e-047 1.42629165e-048 -4.91649348e-049 -1.28304551e-049 -2.06806670e-050 -1.81625833e-051 4.15717978e-052 5.42235144e-054 2.02882921e-053 9.44280346e-054 1.02682495e-055 3.05952356e-055 1.05841663e-055 -3.15209333e-057 4.54793299e-058 -8.61745763e-059 -6.50534437e-059 1.58069278e-059 3.89601629e-060 5.61694295e-061 -3.79889248e-062 -1.84055455e-062 -8.09472280e-063 1.26921944e-064 -2.23213065e-064 6.47004166e-065 3.86738861e-066 -8.22718125e-067 -2.69036211e-067 -2.62337568e-067 -4.68876196e-070 -2.84310868e-069 -8.11379314e-071 4.78135868e-071 2.21861865e-072 -6.09438545e-072 -1.11505706e-072 6.27328556e-074 -2.10136347e-074 1.21689738e-075 1.32236307e-075 4.74897217e-076 -6.65228703e-078 -1.05220462e-077 1.78421294e-078 -1.08709230e-079 1.90437006e-079 2.52935420e-080 1.05042744e-081 9.29038759e-082 -1.75042652e-082 7.94855171e-084 -6.50687293e-084 7.70009774e-085 2.44358524e-085 -9.87174317e-087 6.72852748e-087 -1.69075478e-087 -2.65670189e-088 -1.67339748e-089 5.32786198e-090 -3.09998707e-090 2.11202361e-092 7.83359213e-092 7.08808795e-093 3.30963314e-093 4.81269294e-095 2.71754716e-095 -4.91797614e-096 6.75985046e-096 -9.61376714e-097 -1.68039112e-097 1.27108566e-099 -2.17824803e-098 1.23237345e-100 2.11848816e-100 -6.38251573e-101 -3.24885340e-102 1.74720510e-102 3.11118413e-103]
– Plot the function you found and the data from the dataset. Does the fit look good?
Please check Figure 3.

3.2 Best Periodic fit for regression dataset2

– Write optimization problem to find the coefficients a k and b k .(Please check Figure 4.)
– Find the best fit for the data stored in the file regression dataset 2 (T=1). Does the fit look good?(Please check Figure 5.)

$$\min_{a_0 \cdots a_n} \sum_{i=0}^{N-1} \left(\sum_{k=0}^{K} a_k \cos(kT2\pi x_i) + b_k \sin(kT2\pi x_i) - y_i \right)^2$$

Figure 4: Optimization problem of periodic fit. Coverted this to block matrix form. Please check jupyter notebook to check how I tackled this problem

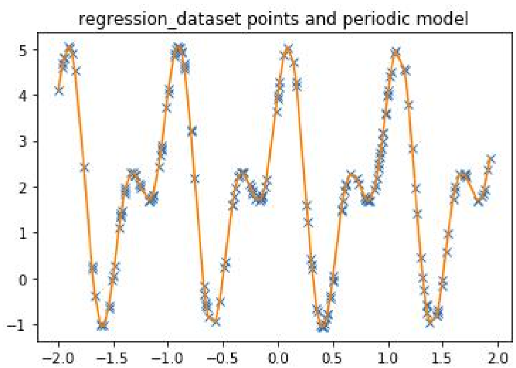

Figure 5: This shows good fit for periodic function and the fitting error is around 0 at degree3 and T=1.