Exercise 1 [Convex optimization with linear equalities]

Consider the following optimization problem

- The Lagrangian of the optimization problem is :

$$\mathcal{L} = \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} + \lambda^T(\mathbf{A}\mathbf{x} - \mathbf{b}) \tag{1}$$

  The KKT conditions for optimality are:

$$\mathbf{Q}\mathbf{x} + \mathbf{A}^T\lambda = 0 \tag{2}$$
$$\mathbf{A}\mathbf{x} - \mathbf{b} = 0 \tag{3}$$

- From equation (1), we have:

$$\mathbf{x} = -\mathbf{Q}^{-1}\mathbf{A}^T\lambda \tag{4}$$

  and substitute this into equation (2), then we have :

$$-\mathbf{A}\mathbf{Q}^{-1}\mathbf{A}^T\lambda = \mathbf{b} \tag{5}$$

$$\Rightarrow \lambda = -(\mathbf{A}\mathbf{Q}^{-1}\mathbf{A}^T)^{-1}\mathbf{b} \tag{6}$$
$$\Rightarrow \mathbf{x} = \mathbf{Q}^{-1}\mathbf{A}^T(\mathbf{A}\mathbf{Q}^{-1}\mathbf{A}^T)^{-1}\mathbf{b} \tag{7}$$

- In this question, $\mathbf{A} = [111]$, $\mathbf{Q} = \begin{bmatrix} 10 & 1 & 0 \\ 1 & 100 & 2 \\ 0 & 2 & 1 \end{bmatrix}$, $\mathbf{b} = 1$,
  Use the equations above, we have:

$$\mathbf{x}^T = [\ 0.09090909, -0.01030928, 0.91940019\ ] \tag{8}$$
$$\lambda = -0.89878163 \tag{9}$$

Exercise 2 [Netwon's method]

- See the attached Jupyter file : Newton's Method.ipynb
- Use the algorithm to compute the minimum of the functions and compare the convergence results with the gradient descent algorithm:
  (1) $f(x) = \mathrm{e}^{\frac{x^2}{100}}$ with $x_0 = 6.5$ as an initial guess. Using the Newton's method, we have:

  (2)$f(\mathbf{x}) = \mathbf{x}^T \begin{bmatrix} 10 & 1 \\ 1 & 10 \end{bmatrix} \mathbf{x}$ with $\mathbf{x}_0 = [-9, -9]]$ as an initial guess. Using the Newton's method, we have:

And we can see the convergence of Newton's Method is much faster than the Gradient method.

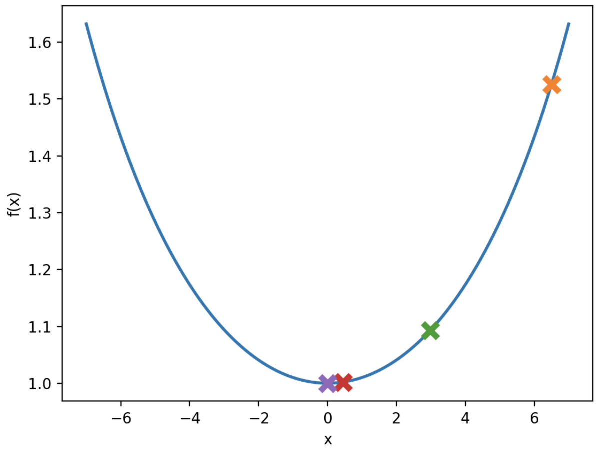


Figure 1: *Find the minimum of f(x).*

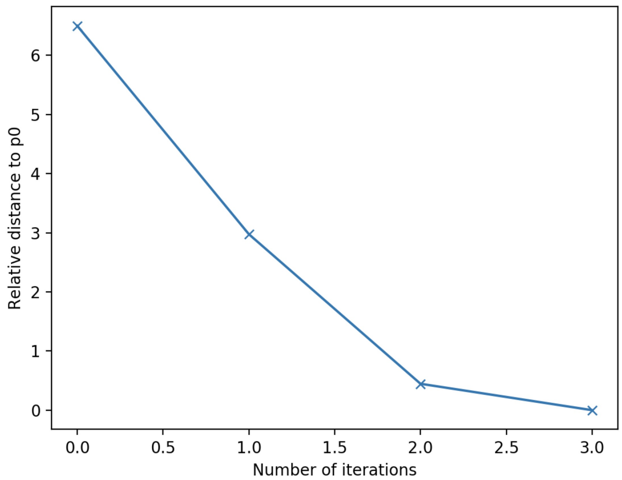


Figure 2: *Times of iteration.*

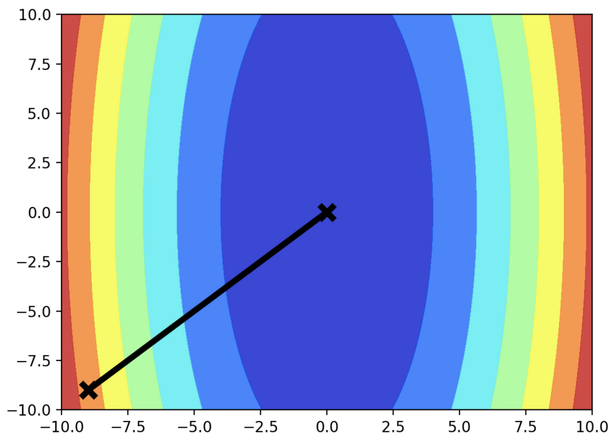


Figure 3: *Find the minimum of f(x).*

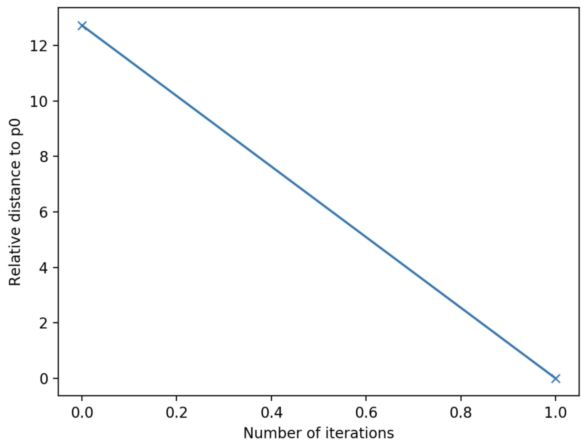
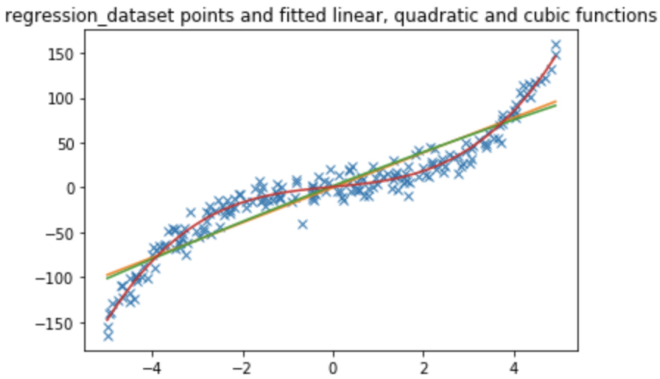


Figure 4: *Times of iteration.*

Exercise 3 [Linear Least Squares]

- Question 1
  - The best order that fits the data is 3. Since we can see the fitting error does not change that much after the order of 3. Thus, higher order is not necessary and may cause overfit.
  - The polynomial coefficient: $\mathbf{a_0} = 0.720251, \mathbf{a_1} = 4.678052, \mathbf{a_2} = 0.055079, \mathbf{a_3} = 1.013701$



  -

- Question 2

$$y = a_0 + \sum_{k=1}^{K} a_k \cos(kT2\pi x_k) + b_k \sin(kT2\pi x_k) \tag{10}$$

  - The optimization problem can be written as

$$\min_{a_0, a_1, a_2, \ldots, a_k, b_1, b_2, \ldots, b_k} \| \mathbf{y} - \hat{\mathbf{y}} \|^2 \tag{11}$$

$$\hat{\mathbf{y}} = \Phi\left(\mathbf{x}\right) \omega \tag{12}$$

$$\tag{13}$$

$$\Phi\left(\mathbf{x}\right)=\begin{bmatrix}1 & cos2T\pi x_0 & ... & cos2KT\pi x_0 & sin2T\pi x_0 & sin4T\pi x_0 & ... & sin2KT\pi x_0\\ 1 & cos2T\pi x_1 & ... & cos2KT\pi x_1 & sin2T\pi x_1 & sin4T\pi x_1 & ... & sin2KT\pi x_1\\ . & . & . & . & . & . & . & .\\ . & . & . & . & . & . & . & .\\ . & . & . & . & . & . & . & .\\ 1 & cos2T\pi x_{n-1} & ... & cos2KT\pi x_{n-1} & sin2T\pi x_{n-1} & sin4T\pi x_{n-1} & ... & sin2KT\pi x_{n-1}\end{bmatrix} \tag{14}$$

$$\omega^T=\left[a_0,a_1,a_2,...,a_k,b_1,b_2,...,b_k\right] \tag{15}$$

The original problem can be written as

$$\min_{\omega}(\Phi\omega-Y)^T(\Phi\omega-Y)$$

which is equal to

$$\min_{\omega}\omega^T\Phi^T\Phi\omega-2Y^T\Phi\omega+Y^2$$

$$\omega=\left(\Phi^T\Phi\right)^{-1}\Phi^TY \tag{16}$$

- See the attached Jupyter file : Linear Least Square Problems Solution.ipynb
- Given T=1:
  K=1, the fitting error is 236.997620
  K = 2, the fitting error is 0.000000
  And when K=3 and higher, the fitting error is 0. Hence, K=2 is the optimal choice.
- Plot the function :