Basic Exercises for Scientific Programming

Hongzhou Ye

January 1, 2018

Abstract

Scientific programming involves many topics of applied math, varying from basic linear algebra to optimization algorithms. This file is a collection of several basic exercises that help you practice some of the essential skills.

1 Linear least square fitting

In the most general sense, the task of least square fitting is to find an approximate mapping $\tilde{f}$ to a given mapping

$$\begin{aligned} f: \mathbb{X} &\to \mathbb{Y} \\ \boldsymbol{x} &\to \boldsymbol{y} = f(\boldsymbol{x}) \end{aligned} \tag{1}$$

by minimizing the least square loss

$$\mathcal{L} = \sum_i \|\tilde{f}(\boldsymbol{x}^{(i)}) - \boldsymbol{y}^{(i)}\|_2^2 \tag{2}$$

based on a set of known data $\{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}$, where

$$\|\boldsymbol{a}\|_2 = \sqrt{\boldsymbol{a}\cdot\boldsymbol{a}} = \left(\sum_\mu a_\mu^2\right)^{1/2} \tag{3}$$

is the 2-norm of vector $\boldsymbol{a}$.

As a special case, linear least square fitting assumes a linear functional form for the approximate mapping $\tilde{f}$, i.e.

$$\tilde{f}(\boldsymbol{x}) = \mathbf{A}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{b}, \tag{4}$$

where $\mathbf{A}$ and $\boldsymbol{b}$ are determined from experimental data. In this exercise, we are going to explore the properties of linear least square fitting.

1. First let us consider an even simpler case, $\boldsymbol{b} = \mathbf{0}$. Show that $\mathbf{A}$ is determined by the following equation

$$\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{A} = \mathbf{X}\mathbf{Y}^{\mathrm{T}} \tag{5}$$

where

$$\mathbf{X} = [\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(N)}], \quad \mathbf{Y} = [\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \cdots, \boldsymbol{y}^{(N)}]. \tag{6}$$

Hint: $\mathcal{L}$ is a function of $\mathbf{A}$

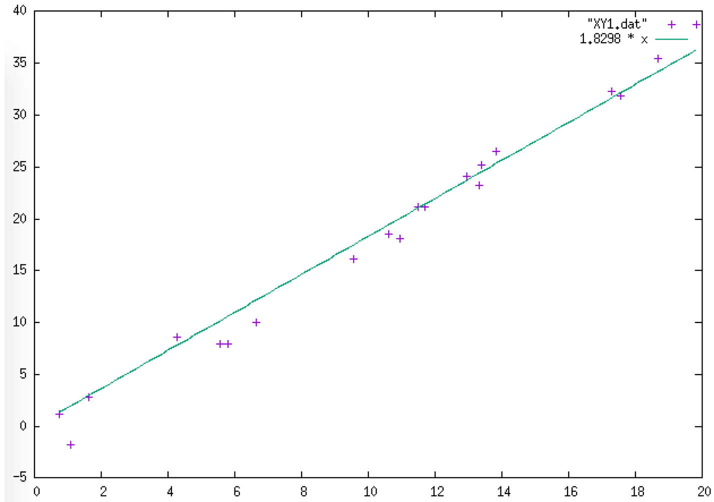
$$\mathcal{L} = \sum_i \|\mathbf{A}^{\mathrm{T}}\boldsymbol{x}^{(i)} - \boldsymbol{y}^{(i)}\|_2^2 = \sum_i (\mathbf{A}^{\mathrm{T}}\boldsymbol{x}^{(i)} - \boldsymbol{y}^{(i)})^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\boldsymbol{x}^{(i)} - \boldsymbol{y}^{(i)}) \tag{7}$$

You might need the following trick

$$\begin{aligned} \sum_i \boldsymbol{x}^{(i)\mathrm{T}}\mathbf{A}\mathbf{A}^{\mathrm{T}}\boldsymbol{x}^{(i)} &= \sum_i \sum_{\mu\nu\lambda} x_\mu^{(i)} A_{\mu\nu} A_{\nu\lambda}^{\mathrm{T}} x_\lambda^{(i)} = \sum_{\mu\nu\lambda} A_{\nu\lambda}^{\mathrm{T}} \left[\sum_i x_\lambda^{(i)} x_\mu^{(i)}\right] A_{\mu\nu} \\ &= \sum_{\mu\nu\lambda} A_{\nu\lambda}^{\mathrm{T}} (XX^{\mathrm{T}})_{\lambda\mu} A_{\mu\nu} = \mathbf{A}^{\mathrm{T}}\mathbf{X}\mathbf{X}^{\mathrm{T}}\mathbf{A} \end{aligned} \tag{8}$$

Same trick can be played to all other terms in the expansion. Then you can take derivative with $\mathbf{A}^{\mathrm{T}}$ and obtain the desired equation (you can treat $\mathbf{A}$ and $\mathbf{A}^{\mathrm{T}}$ two independent variables).

2. We now have the equation and let us see how it works! In `data/least_square` you can find the data file `X1.txt` and `Y1.txt`, which contains 20 data points. Fit a linear equation with zero $\boldsymbol{b}$ and plot the fitted curve along with the original data. Your result should be something like the following



3. The example above is one-dimensional and trivial. A non-trivial multi-dimensional data set can be found in `X2.txt` and `Y2.txt`, where $\boldsymbol{x}^{(i)} \in \mathbb{R}^5$ and $\boldsymbol{y}^{(i)} \in \mathbb{R}^3$ and there are 20 of them. Find an $\mathbf{A} \in \mathbb{R}^{5\times3}$ that minimizes the square loss. If you do the math correctly, the least square loss as defined in eqn (7) should be 3.431402129330746.