

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

$$\begin{aligned} f : \mathbb{X} &\rightarrow \mathbb{Y} \\ \mathbf{x} &\rightarrow \mathbf{y} = f(\mathbf{x}) \end{aligned} \quad (1)$$

by minimizing the least square loss

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

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

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

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

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

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

where \mathbf{A} and \mathbf{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, $\mathbf{b} = \mathbf{0}$. Show that \mathbf{A} is determined by the following equation

$$\mathbf{X}\mathbf{X}^T \mathbf{A} = \mathbf{X}\mathbf{Y}^T \quad (5)$$

where

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

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

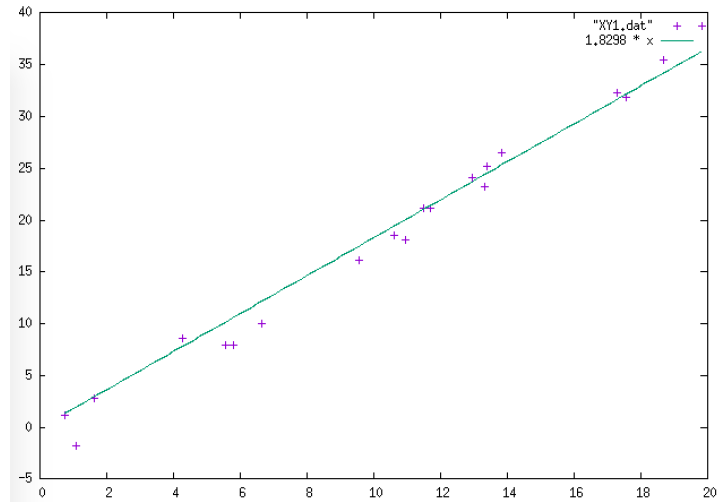
$$\mathcal{L} = \sum_i \|\mathbf{A}^T \mathbf{x}^{(i)} - \mathbf{y}^{(i)}\|_2^2 = \sum_i (\mathbf{A}^T \mathbf{x}^{(i)} - \mathbf{y}^{(i)})^T (\mathbf{A}^T \mathbf{x}^{(i)} - \mathbf{y}^{(i)}) \quad (7)$$

You might need the following trick

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

Same trick can be played to all other terms in the expansion. Then you can take derivative with \mathbf{A}^T and obtain the desired equation (you can treat \mathbf{A} and \mathbf{A}^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 \mathbf{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 $\mathbf{x}^{(i)} \in \mathbb{R}^5$ and $\mathbf{y}^{(i)} \in \mathbb{R}^3$ and there are 20 of them. If you do the math correctly, the least square loss as defined in eqn (7) should be 3.431402129330746.