EECS 127/227AT Optimization Models in Engineering UC Berkeley Spring 2024

# Discussion 5

1. **Product Rule via Chain Rule (10 pts)**

Let $f, g, h\colon \mathbb{R}^n \to \mathbb{R}$ be continuously differentiable functions, such that

$$f(\vec{x}) = g(\vec{x})h(\vec{x}), \qquad \forall \vec{x} \in \mathbb{R}^n. \tag{1}$$

Using the vector-valued chain rule, prove that

$$\nabla f(\vec{x}) = h(\vec{x})\nabla g(\vec{x}) + g(\vec{x})\nabla h(\vec{x}). \tag{2}$$

*HINT: Try to write $f$ as a composition of two functions.*

**Solution:** We write

$$f = f_1 \circ f_2, \qquad \text{where} \quad f_1(a, b) = a \cdot b \quad \text{and} \quad f_2(\vec{x}) = \begin{bmatrix} g(\vec{x}) \\ h(\vec{x}) \end{bmatrix}. \tag{3}$$

Thus

$$\begin{aligned} Df(\vec{x}) &= D(f_1 \circ f_2)(\vec{x}) \qquad &(4)\\ &= [Df_1(f_2(\vec{x}))][Df_2(\vec{x})]. \qquad &(5) \end{aligned}$$

The individual Jacobians are

$$\begin{aligned} Df_1(a, b) &= \begin{bmatrix} \frac{\partial f_1}{\partial a}(a, b) & \frac{\partial f_1}{\partial b}(a, b) \end{bmatrix} \qquad &(6)\\ &= \begin{bmatrix} b & a \end{bmatrix}. \qquad &(7)\\ Df_2(\vec{x}) &= \begin{bmatrix} Dg(\vec{x}) \\ Dh(\vec{x}) \end{bmatrix} \qquad &(8)\\ &= \begin{bmatrix} [\nabla g(\vec{x})]^\top \\ [\nabla h(\vec{x})]^\top \end{bmatrix}. \qquad &(9) \end{aligned}$$

Thus

$$\begin{aligned} Df(\vec{x}) &= [Df_1(f_2(\vec{x}))][Df_2(\vec{x})] \qquad &(10)\\ &= \begin{bmatrix} h(\vec{x}) & g(\vec{x}) \end{bmatrix} \begin{bmatrix} [\nabla g(\vec{x})]^\top \\ [\nabla h(\vec{x})]^\top \end{bmatrix} \qquad &(11)\\ &= h(\vec{x})[\nabla g(\vec{x})]^\top + g(\vec{x})[\nabla h(\vec{x})]^\top \qquad &(12)\\ \nabla f(\vec{x}) &= h(\vec{x})\nabla g(\vec{x}) + g(\vec{x})\nabla h(\vec{x}). \qquad &(13) \end{aligned}$$

2. **Linear Regression Versus Orthogonal Distance Regression**

In this exercise, we explore three regression techniques:

- ordinary least squares (OLS), a formulation for solving linear regression,

- orthogonal distance regression (ODR), solved using PCA, and
- total least squares (TLS), a generalization of OLS where observation error on both the dependent and independent variables is assumed

We will examine these regression methods in turn and compare their possible use cases.

In general, regression is used to model the relationship between observed input data and corresponding output data. In this problem, we consider $n$ input data points $\vec{a}_i \in \mathbb{R}^d$ and $n$ corresponding output data points $b_i \in \mathbb{R}$. Note that the input comprises $d$ real-valued features and the output is a real-valued scalar.

In the case of *linear regression* (LR), each output $b_i$ is assumed to be approximately a linear combination of the features of the input $\vec{a}_i$, i.e., $b_i \approx \vec{a}_i^\top \vec{x}$, where $\vec{x} \in \mathbb{R}^d$ is a $d$–dimensional vector of weights used in the linear combination. We define the LR computation as finding the $\vec{x}$ that minimizes the sum of the squared errors between the outputs $b_i$ and the predicted outputs $\vec{a}_i^\top \vec{x}$ (least-squares), i.e., computing

$$\vec{x}^\star_{\mathrm{LR}} = \operatorname*{argmin}_{\vec{x}} \sum_{i=1}^{n} (\vec{a}_i^\top \vec{x} - b_i)^2. \tag{14}$$

Assume for the entirety of this problem that the data are centered, i.e., for all $j \in \{1, \dots, d\}$, we have $\sum_{i=1}^n a_{ij} = 0$ and $\sum_{i=1}^n b_i = 0$. This means means that all trendlines we compute (during LR, and later, ODR) will pass through the origin.

(a) Show that the LR computation can be formulated as a least squares problem of the form

$$\vec{x}^\star_{\mathrm{LR}} = \operatorname*{argmin}_{\vec{x}} \left\| A\vec{x} - \vec{b} \right\|_2^2, \tag{15}$$

where

$$A = \begin{bmatrix} \leftarrow \vec{a}_1^\top \rightarrow \\ \vdots \\ \leftarrow \vec{a}_i^\top \rightarrow \\ \vdots \\ \leftarrow \vec{a}_n^\top \rightarrow \end{bmatrix} \qquad \vec{b} = \begin{bmatrix} b_1 \\ \vdots \\ b_i \\ \vdots \\ b_n \end{bmatrix}. \tag{16}$$

**Solution:** We have

$$\begin{aligned} \left\| A\vec{x} - \vec{b} \right\|_2^2 &= \left\| \begin{bmatrix} \vec{a}_1^\top \\ \vdots \\ \vec{a}_n^\top \end{bmatrix} \vec{x} - \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \right\|_2^2 \\ &= \left\| \begin{bmatrix} \vec{a}_1^\top \vec{x} \\ \vdots \\ \vec{a}_n^\top \vec{x} \end{bmatrix} - \begin{bmatrix} b_1 \\ \vdots \\ b_n \end{bmatrix} \right\|_2^2 \\ &= \left\| \begin{bmatrix} \vec{a}_1^\top \vec{x} - b_1 \\ \vdots \\ \vec{a}_n^\top \vec{x} - b_n \end{bmatrix} \right\|_2^2 \\ &= \sum_{i=1}^{n} (\vec{a}_i^\top \vec{x} - b_i)^2. \end{aligned}$$

(b) We now consider an example LR computation in which $d = 1$ and $n = 3$. Let

$$A = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \qquad \vec{b} = \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix}. \tag{17}$$

Compute the best fit LR regression line — in this case, a 1–dimensional scalar representing the slope of the line — $x^\star_{\mathrm{LR}}$.

**Solution:** We use the least squares formula $x^\star_{\mathrm{LR}} = (A^\top A)^{-1}A^\top\vec{b}$. We have

$$A^\top A = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} = 2 \tag{18}$$

$$\begin{aligned} x^\star_{\mathrm{LR}} &= (A^\top A)^{-1}A^\top\vec{b} && (19) \\ &= \frac{1}{2}A^\top\vec{b} && (20) \\ &= \frac{1}{2}\begin{bmatrix} -1 & 0 & 1 \end{bmatrix}\begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix} && (21) \\ &= \frac{3}{2}. && (22) \end{aligned}$$

(c) Now let

$$A = \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix} \qquad \vec{b} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}. \tag{23}$$

Compute the best fit LR regression value $x^\star_{\mathrm{LR}}$.

**Solution:**

$$A^\top A = \begin{bmatrix} -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix} = 6 \tag{24}$$

$$\begin{aligned} x^\star_{\mathrm{LR}} &= (A^\top A)^{-1}A^\top\vec{b} && (25) \\ &= \frac{1}{6}A^\top\vec{b} && (26) \\ &= \frac{1}{6}\begin{bmatrix} -1 & -1 & 2 \end{bmatrix}\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} && (27) \\ &= \frac{3}{6} = \frac{1}{2}. && (28) \end{aligned}$$

(d) Note that the computations in (b) and (c) above are performed on the same values; inputs and outputs are simply switched. Plot these data points on the $a$–$b$ plane and plot the trendlines corresponding to each $x^\star_{\mathrm{LR}}$ from (b) and (c) above. Are these trendlines the same? Reason geometrically about why this is the case.

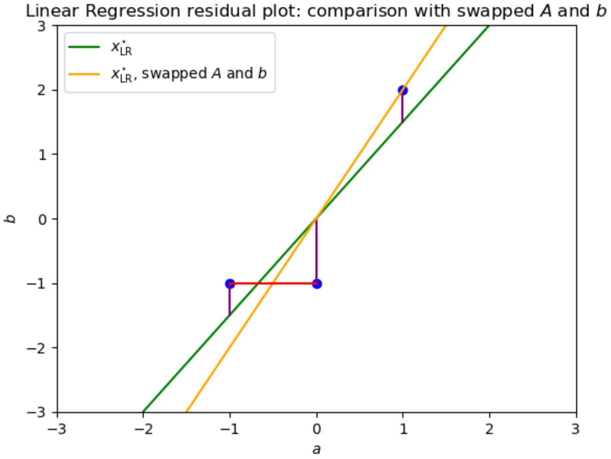

**Figure 1:** Illustration of solution of (b) and (c) on the $a - b$ plane.

**Solution:** See figure 1: in the case where $d = 1$, the LR in (b) finds the line that goes through the origin and minimizes the sum of the squares of the vertical distance ($y$-distance) from the points to the line. The LR in (c) finds the line that goes through the origin and minimizes the sum of the squares of the horizontal distance ($x$-distance).

In some cases, we may not have a preference for which values are designated inputs or outputs and want to avoid differences in regression result based on our choice. In this case, we can use *orthogonal distance regression* (ODR) to compute the regression line that minimizes the sum of the squares of the orthogonal distances of each data point to the line.

To perform the ODR computation, we define vectors $\vec{z}_i$, each of which concatenates all inputs and outputs at observation point $i$, i.e.,

$$\vec{z}_i = \begin{bmatrix} \uparrow \\ \vec{a}_i \\ \downarrow \\ b_i \end{bmatrix}. \tag{29}$$

The ODR regression line is then the direction $\vec{x} \in \mathbb{R}^{d+1}$ such that the sum of squares of the (orthogonal) distances between the points $\vec{z}_i$ and their projections on the line passing through the origin along direction $\vec{x}$ are minimized.

$$\vec{x}^\star_{\mathrm{ODR}} = \operatorname*{argmin}_{\vec{x}:\|\vec{x}\|_2=1} \sum_{i=1}^{n} \left\| \vec{z}_i - \mathrm{proj}_{\mathrm{span}(\vec{x})}(\vec{z}_i) \right\|_2^2 \tag{30}$$

$$= \operatorname*{argmin}_{\vec{x}:\|\vec{x}\|_2=1} \sum_{i=1}^{n} \min_{v_i} \|\vec{z}_i - v_i\vec{x}\|_2^2 . \tag{31}$$

(e) Show that the ODR computation can be formulated as the problem of finding the eigenvector corresponding to the largest eigenvalue of $H = \sum_{i=1}^{n} \vec{z}_i\vec{z}_i^\top$, i.e.,

$$\vec{x}^\star_{\mathrm{ODR}} = \operatorname*{argmax}_{\vec{x}:\|\vec{x}\|_2=1} \vec{x}^\top H\vec{x}. \tag{32}$$

Solve ODR using the singular value decomposition (SVD) of the "augmented" data matrix:

$$Z = \begin{bmatrix} \leftarrow \vec{z}_1^\top \rightarrow \\ \vdots \\ \leftarrow \vec{z}_i^\top \rightarrow \\ \vdots \\ \leftarrow \vec{z}_n^\top \rightarrow \end{bmatrix}. \tag{33}$$

**Solution:** The inner minimization problem can be solved using projections to obtain,

$$\mathrm{proj}_{\mathrm{span}(\vec{x})}(\vec{z}_i) = \underset{v_i}{\mathrm{argmin}} \|\vec{z}_i - v_i \vec{x}\|_2^2 = \vec{z}_i^\top \vec{x} \tag{34}$$

Substituting this into the expression from original definition of ODR we have,

$$\sum_{i=1}^{n} \left\|\vec{z}_i - (\vec{z}_i^\top \vec{x})\vec{x}\right\|_2^2 = \sum_{i=1}^{n} \vec{z}_i^\top \vec{z}_i - 2\vec{x}^\top \vec{z}_i \vec{z}_i^\top \vec{x} + \vec{x}^\top \vec{z}_i \vec{z}_i^\top \vec{x}\vec{x}^\top \vec{x} \tag{35}$$

$$= \left(\sum_{i=1}^{n} \vec{z}_i^\top \vec{z}_i\right) - \left(\sum_{i=1}^{n} \vec{x}^\top \vec{z}_i \vec{z}_i^\top \vec{x}\right) \tag{36}$$

$$= \left(\sum_{i=1}^{n} \vec{z}_i^\top \vec{z}_i\right) - \vec{x}^\top \left(\sum_{i=1}^{n} \vec{z}_i \vec{z}_i^\top\right) \vec{x} \tag{37}$$

$$\underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmin}} \sum_{i=1}^{n} \min_{v_i} \|\vec{z}_i - v_i \vec{x}\|_2^2 = \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmin}} \left(\sum_{i=1}^{n} \vec{z}_i^\top \vec{z}_i\right) - \vec{x}^\top \left(\sum_{i=1}^{n} \vec{z}_i \vec{z}_i^\top\right) \vec{x} \tag{38}$$

$$= \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \vec{x}^\top \left(\sum_{i=1}^{n} \vec{z}_i \vec{z}_i^\top\right) \vec{x} \tag{39}$$

$$= \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \vec{x}^\top H \vec{x} \tag{40}$$

Note that the second line follows because we enforce in our optimization that $\vec{x}$ is unit norm, and thus $\vec{x}^\top \vec{x} = 1$. The solution is the eigenvector corresponding to the largest eigenvalue of $H$:

$$\underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \vec{x}^\top H \vec{x} = \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \vec{x}^\top U \Lambda U^\top \vec{x} \tag{41}$$

$$= \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}} \sum_{j=1}^{d+1} (\vec{x}^\top \vec{u}_i) \lambda_i (\vec{u}_i^\top \vec{x}) \tag{42}$$

$$\leq \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \lambda_1 \sum_{j=1}^{d+1} (\vec{x}^\top \vec{u}_i)(\vec{u}_i^\top \vec{x}) \tag{43}$$

$$= \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \lambda_1 \vec{x}^\top U U^\top \vec{x} \tag{44}$$

$$= \underset{\vec{x}, \|\vec{x}\|_2 = 1}{\mathrm{argmax}}\, \lambda_1 \left\|U^\top \vec{x}\right\|^2 \tag{45}$$

$$= \lambda_1. \tag{46}$$

We know that $\vec{u}_1^\top H \vec{u}_1 = \lambda_1$, so $\vec{u}_1 \in \mathrm{argmax}_{\vec{x}, \|\vec{x}\|_2 = 1}\, \vec{x}^\top H \vec{x}$.

Note that $H = Z^\top Z$, so $u_1$ is also a right singular vector of $Z$ associated with the largest singular value of $Z$.

(f) Considering $d = 1$ and $n = 3$, numerically find the results of the ODR when

$$A = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \qquad \vec{b} = \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix}. \tag{47}$$

using the associated Jupyter notebook. Are the results similar if the $A$ and $\vec{b}$ values are switched?

**Solution:** See the associated notebook for the code generating the ODR.

$$\vec{x}^{\star}_{\mathrm{ODR}} \approx \begin{bmatrix} 0.472 \\ 0.882 \end{bmatrix}. \tag{48}$$

In the case that inputs $A$ and $b$ are switched, the notebook returns the same output.

(g) Compare the two techniques. If your goal is to understand a invertible symmetric relationship between the "Parent height" and the "Child height", which method would you prefer? See figure 2.

**Solution:** ODR has the nice property that it treats the "measurement matrix" and the "measurements" symmetrically — so unlike LR, it does not give you two different answers when you switch the roles of $\vec{a}$ and $\vec{b}$, in a simple scalar regression case. So if you really trust your "experiment" i.e. the values in the matrix $A$ then you can use LR, but if you want to just understand the relationship between the $(a_i, b_i)$ pairs in a symmetric way, then ODR will help with that. In the case of figure 2, since you want to be able to just understand the relationships between the two heights, ODR might be a better choice.

Orthogonal Distance Regression assumes that there is no causal relationship between the two variables, and finds a relationship between two variables that explains the most variance (hence why we solve it using PCA). It is often, however, the case that we seek to find a causal relationship while also acknowledging that there is some measurement error in our dependent variable. Suppose again that we are seeking a relationship of the form $A\vec{x} \approx \vec{y}$. Instead of finding some $\vec{x}$ such that $\hat{\vec{y}} = A\vec{x}$ and $\|\vec{y} - \hat{\vec{y}}\|_2^2$ is minimized (as in OLS), we can seek to find $\tilde{A}$ and $\tilde{\vec{y}}$ such that $\tilde{\vec{y}} = \tilde{A}\vec{x}$ for some $\vec{x}$ and

$$\|A - \tilde{A}\|_F^2 + \|\vec{y} - \tilde{\vec{y}}\|_2^2$$

is minimized. Total Least Squares (TLS) solves this problem, equivalently expressed below:

$$\min_{\tilde{A}, \tilde{\vec{y}}} \left\| \begin{bmatrix} A & \vec{y} \end{bmatrix} - \begin{bmatrix} \tilde{A} & \tilde{\vec{y}} \end{bmatrix} \right\|_F^2 \tag{49}$$

$$\text{s.t.} \quad \begin{bmatrix} \tilde{A} - \tilde{\vec{y}} \end{bmatrix} \begin{bmatrix} \vec{x} \\ -1 \end{bmatrix} = \vec{0} \tag{50}$$

(h) In the case where

$$A = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \quad \vec{x} = x \in \mathbb{R} \quad \vec{y} = \begin{bmatrix} -1 \\ -1 \\ 2 \end{bmatrix}$$

how would we find $\begin{bmatrix} \tilde{A} & \tilde{\vec{y}} \end{bmatrix}$ as well as the corresponding $x$? Solve for the matrix $\begin{bmatrix} \tilde{A} & \tilde{\vec{y}} \end{bmatrix}$ and $x$ in the associated jupyter notebook. How do these results compare to OLS and ODR?

*HINT: What must the rank of* $\begin{bmatrix} \tilde{A} & \tilde{\vec{y}} \end{bmatrix}$ *be? How can we find the best such approximation?*

**Solution:** See associated jupyter notebook.

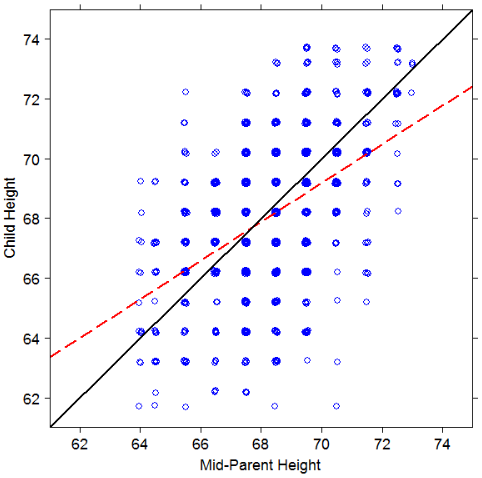

**Figure 2:** Illustration of the correlation between the heights of adults and their parents. In red is the result of linear regression. In black is the result of the orthogonal direction regression. This work has first been done by Galton in 1886. Using linear regression, Galton remarks that: "It appeared from these experiments that the offspring did not tend to resemble their parents in size, but always to be more mediocre than they – to be smaller than the parents, if the parents were large; to be larger than the parents, if the parents were small." Figure comes from https://select-statistics.co.uk/blog/regression-to-the-mean-as-relevant-today-as-it-was-in-the-1900s/ The original text can be found at https://www.jstor.org/stable/2841583.