1. Quadratic Approximation and Vector Differentiation

As shown in the previous discussion, a common way to locally approximate a non-linear high-dimensional functions is to perform linearization near a point. In the case of a two-dimensional function \( f(x, y) \) with scalar output, the linear approximation of \( f(x, y) \) at a point \((x_\star, y_\star)\) is given by

\[
f(x, y) \approx f(x_\star, y_\star) + \frac{\partial f}{\partial x}(x - x_\star) + \frac{\partial f}{\partial y}(y - y_\star)
\]  

(1)

In vector form, this can be written as:

\[
f(\vec{x}) \approx f(\vec{x}_\star) + [D_{\vec{x}}f]|_{\vec{x}_\star} (\vec{x} - \vec{x}_\star).
\]  

(2)

Recall from the previous discussion that \( D_{\vec{x}}f \) is a row-vector filled with the partial derivatives \( \frac{\partial f(\vec{x})}{\partial x_i} \):

\[
D_{\vec{x}}f = \begin{bmatrix} \frac{\partial f(\vec{x})}{\partial x_1} & \cdots & \frac{\partial f(\vec{x})}{\partial x_n} \end{bmatrix}.
\]  

(3)

Our goal is to extend this idea to a quadratic approximation. To do this, we need some notion of a second derivative. For this discussion, we will only be considering functions from \( \mathbb{R}^n \rightarrow \mathbb{R} \), since that is the typical form for a cost function used during optimization.

Why could a quadratic approximation be useful? Consider the plot from last week’s discussion, below, which has been annotated with linear and quadratic approximations (the second of which we will learn to calculate). The quadratic approximation tracks the original curve more closely, reducing the error at larger values of \( \delta \) away from \( x_\star \).
(a) **Given the function** \( f(x) = e^{-2x} \), **find the first and second derivatives**, and **write out its quadratic approximation at** \( x = x^* \). **Hint:** Use a Taylor series expansion.

(b) Given a multivariable function, we can take second partial derivatives. The first derivative can take the partial derivative with respect to \( x \) or \( y \). For each of these first derivatives, we can again take a partial derivative on \( x \) or \( y \), yielding 4 total second partial derivatives. For example, the notation below indicates taking a partial on \( x \), then a partial on \( y \) (for multiple partials, we read/apply the denominator right-to-left, or inside-to-outside by convention).

\[
\frac{\partial}{\partial y} \left( \frac{\partial f}{\partial x} \right) = \frac{\partial^2 f}{\partial y \partial x}.
\]

**Given the function** \( f(x, y) = x^2y^2 \), **find all of the first and second partial derivatives**. Specifically, these are \( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial^2 f}{\partial x \partial x}, \frac{\partial^2 f}{\partial x \partial y}, \frac{\partial^2 f}{\partial y \partial x}, \frac{\partial^2 f}{\partial y \partial y} \).

(c) To find the quadratic approximation of \( f(x, y) \) near \((x^*, y^*)\), we **plug in** \( f(x^* + \Delta x, y^* + \Delta y) \) and **drop the terms that are higher order than quadratic**. Notice that \( \Delta \) here means the same thing as \( \delta \) did in previous discussions (this can change purely depending on preference.)

\[
f(x^* + \Delta x, y^* + \Delta y) = (x^* + \Delta x)^2(y^* + \Delta y)^2
\]

\[
= (x^2 + 2x^* \Delta x + (\Delta x)^2)(y^2 + 2y^* \Delta y + (\Delta y)^2)
\]

\[
\approx x^2y^2 + 2x^*y^*\Delta x + 2x^2y^*\Delta y
\]

\[
+ y^2(\Delta x)^2 + 4x^*y^*(\Delta x)(\Delta y) + x^2(\Delta y)^2
\]

\[
= f(x^*, y^*) + \frac{\partial f}{\partial x} \Delta x + \frac{\partial f}{\partial y} \Delta y
\]

\[
+ \frac{1}{2} \frac{\partial^2 f}{\partial x^2} (\Delta x)^2 + \frac{1}{2} \frac{\partial^2 f}{\partial y^2} (\Delta y)^2
\]

\[
+ \frac{\partial^2 f}{\partial x \partial y} (\Delta x)(\Delta y).
\]
This is slightly different from the expression we get via the Taylor series expansion. **How would we rewrite this expression, so that all 4 second derivatives are involved, each with a coefficient of \(\frac{1}{2}\)?**

*Hint: what was the relationship we found between \(\frac{\partial^2 f}{\partial x \partial y}\) and \(\frac{\partial^2 f}{\partial y \partial x}\)?*

(d) Just as we created the derivative row vector to hold all the first partial derivatives to help in writing linearization in matrix/vector form:

\[
D\vec{x} f = \left[ \begin{array}{c} \frac{\partial f(\vec{x})}{\partial x_1} \\ \vdots \\ \frac{\partial f(\vec{x})}{\partial x_n} \end{array} \right] \tag{12}
\]

we would like to create a matrix to hold all the second partial derivatives to help in writing quadratic approximation in matrix/vector form:

\[
H\vec{x} f = \left[ \begin{array}{ccc} \frac{\partial^2 f(\vec{x})}{\partial x_1^2} & \cdots & \frac{\partial^2 f(\vec{x})}{\partial x_1 \partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\vec{x})}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 f(\vec{x})}{\partial x_n^2} \end{array} \right] \tag{13}
\]

This matrix is the **Hessian** of \(f\). Note that this quantity is different from the *Jacobian* matrix that was covered in the previous discussion. In contrast to the Hessian, which is the matrix of second partial derivatives of a **scalar-valued vector-input** function \(f: \mathbb{R}^n \to \mathbb{R}\), the Jacobian is the matrix of first partial derivatives of a **vector-valued vector-input** function \(\vec{f}: \mathbb{R}^n \to \mathbb{R}^k\).

In fact, the Hessian is the (Jacobian) derivative of the derivative; if we let \(\vec{g}(\vec{x}) = (D\vec{x} f)^\top\) (so that it’s a column vector and the dimensions work out), then \(H\vec{x} f = D\vec{x} \vec{g}\). **To get a feel for the Hessian of \(f\), find \(H_{(x,y)} f\) for the same \(f\) as above.** \(f(x,y) = x^2 y^2\). [Practice]: For additional practice, try computing the hessian of \(f_2(x, y, z) = x^2 y^2 z + x^3 z^2 + y\).
(e) Using the Hessian, write out the general formula for the quadratic approximation of a scalar-valued function $f$ of a vector $\vec{x}$ in vector/matrix form. That is, we want to have an expression of the following form:

$$f(\vec{x}_* + \Delta \vec{x}) \approx f(\vec{x}_*) + \left[ D_{\vec{x}} f|_{\vec{x}_*} \right] (\Delta \vec{x}) + \underline{\text{___________}} \quad (14)$$

What populates the underlined space?

(f) The second derivative also has an interpretation as the derivative of the derivative. However, we saw that the derivative of a scalar-valued function with respect to a vector is naturally a row. If you wanted to approximate how much the first derivative changed by moving a small amount $\Delta \vec{w}$, how would you get such an estimate using your expression for the second derivative?
(g) [Practice]: Show that the quadratic approximation for the scalar-valued function \( f(\vec{w}) = e^{\vec{x}^T \vec{w}} \) around \( \vec{w} = \vec{w}_* \) is

\[
f(\vec{w}_* + \Delta \vec{w}) \approx e^{\vec{x}^T \vec{w}_*} \left( 1 + \vec{x}^T (\Delta \vec{w}) + \frac{1}{2} \left( \vec{x}^T (\Delta \vec{w}) \right)^2 \right).
\] (15)

assuming that \( \vec{x} \) is just some constant, given vector.

*Hint:* You can compute the following partial derivatives:

\[
\frac{\partial f}{\partial w_i} = x_i f(\vec{w})
\] (16)

\[
\frac{\partial^2 f}{\partial w_j \partial w_i} = x_i x_j f(\vec{w}).
\] (17)

Now compute \( D\vec{w} f \) and \( H\vec{w} f \), and plug it into the quadratic approximation formula.

(h) Using the result in the previous subpart, use linearity to give the quadratic approximation for the function \( \sum_{i=1}^{m} e^{\vec{x}_i^T \vec{w}} \) around \( \vec{w} = \vec{w}_* \). Here, assume that the \( \vec{x}_i \) are just some given vectors.
Contributors:

- Neelesh Ramachandran.
- Druv Pai.
- Pavan Bhargava.
- Anant Sahai.