IBF Applied Mathematics for Machine Learning Part II: Vector Calculus

Yilin Wu

1 Introduction

Many algorithms in machine learning optimize an objective function with respect to a set of desired model parameters that control how well a model explains the data: Finding good parameters can be phrased as an optimization problem. Examples include:

- linear regression, where we look at curve-fitting problems and optimize linear weight parameters to maximize the likelihood;
- neural network auto-encoders for dimensionality reduction and data compression, where the parameters are the weights and biases of each layer, and where we minimize a reconstruction error by repeated application of the chain rule;
- Gaussian mixture models for modeling data distributions, where we optimize the location and shape parameters of each mixture component to maximize the likelihood of the model.

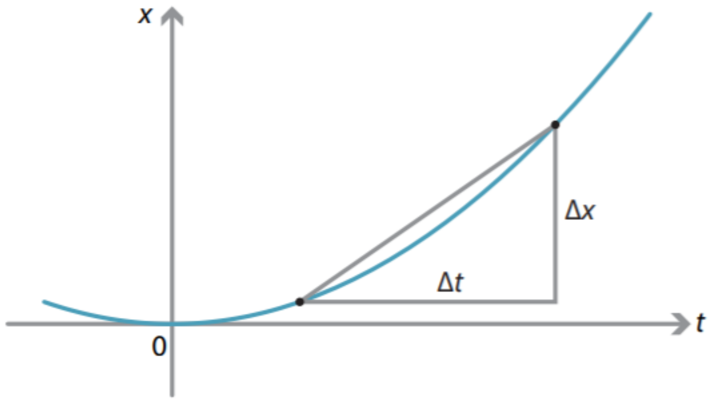
For the above examples, we typically solve by using optimization algorithms that exploit gradient information. Vector calculus will be applied to Optimization, Regression, Dimensionality reduction, density estimation and classification problems.

2 Differentiation of univariate functions

First recall, the formula for **average velocity**

$$v = \frac{\Delta x}{\Delta t}$$

where Δx is the change in your position and Δt is the time taken. Thus, average velocity is the rate of change of position with respect to time. Draw a graph of your position $x(t)$ at time t seconds. Connect two points on the graph, representing your position at two different times. The gradient of this line is your average velocity over that time period. Trying to be more



Average velocity $v = \frac{\Delta x}{\Delta t}$.

accurate, you look at shorter time intervals, with Δt smaller and smaller. If you really knew your position at every single instant of time, then you could work out your average velocity over any time interval, no matter how short. The three lines in the following diagram correspond to $\Delta t = 2$, $\Delta t = 1$ and $\Delta t = 0.5$. As Δt approaches 0, you obtain better and better estimates of your instantaneous velocity at the instant $t = 1$. These estimates correspond to the gradients of lines connecting closer and closer points on the graph.

In the limit, as $\Delta t \to 0$,

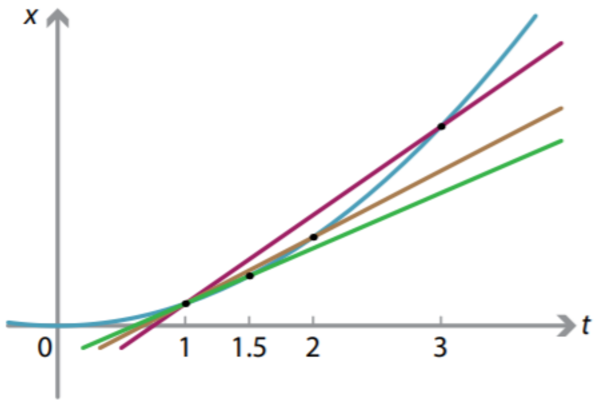
$$\lim_{\Delta t \to 0} \frac{\Delta x}{\Delta t}$$

gives the precise value for the instantaneous velocity at $t = 1$. This is also the gradient of the tangent to the graph at $t = 1$. Instantaneous velocity is the instantaneous rate of change of position with respect to time.

Definition: The derivative (differentiation) of f at x is defined as the limit

$$\frac{df}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h},$$

and the derivative at certain point is also the slope of the tangent line at that point.



Average velocity over shorter time intervals.

The derivative of f points in the direction of steepest ascent of f.

Exercise: Find the derivative of $y = x^2$ at $x = 2$ using definition.

2.1 Differentiation Rules

1. $\frac{d}{dx}(c) = 0$ The constant rule
2. $\frac{d}{dx}(cf(x)) = cf'(x)$ The multiple rule
3. $\frac{d}{dx}(f(x) \pm g(x)) = f'(x) \pm g'(x)$ The sum/difference rule
4. $\frac{d}{dx}(f(x)g(x)) = f'(x)g(x) + f(x)g'(x)$ The product rule
5. $\frac{d}{dx}\left(\frac{f(x)}{g(x)}\right) = \frac{f'(x)g(x) - f(x)g'(x)}{g(x)^2}$ The quotient rule
6. $\frac{d}{dx}f(g(x)) = f'(g(x))g'(x)$ The chain rule
7. $\frac{d}{dx}(x^n) = nx^{n-1}$ The power rule

2.2 Differentiation Table

$\frac{d}{dx}(e^x) = e^x$	$\frac{d}{dx}(a^x) = a^x \ln(a)$
$\frac{d}{dx}(ln(\|x\|)) = \frac{1}{x}$	$\frac{d}{dx}(\sin x) = \cos x$
$\frac{d}{dx}(\cos x) = -\sin x$	$\frac{d}{dx}(\tan x) = \sec^2 x$
$\frac{d}{dx}(\cot x) = -\csc^2 x$	$\frac{d}{dx}(\sec x) = \sec x \tan x$
$\frac{d}{dx}(\csc x) = -\csc x \cot x$	$\frac{d}{dx}(\arcsin x) = \frac{1}{\sqrt{1-x^2}}$
$\frac{d}{dx}(\arctan x) = \frac{1}{1+x^2}$	$\frac{d}{dx}(x^n) = nx^{n-1}$

Exercise: Compute the derivative of the function $h(x) = (2x+1)^4$ using the chain rule.

3 Partial derivatives and gradients

Differentiation discussed in the previous section applies to functions f of a scalar variable. In this section we consider the general case where the function f depends on more than one variables ($f(\vec{x}) = f(x_1, x_2)$). The generalization of the derivative to functions of several variables is the **gradient**. We find the gradient of the function f with respect to $\vec{x}$ by varying one variable at a time and keeping the others constant. The gradient is then the collection of these **partial derivatives**.

Exercise: Find the partial derivative for $f(x, y) = (x + 2y^3)^2$.

We usually define the gradient vector as a column vectors.
Exercise: Find the gradient of $f(x_1, x_2) = x_1^2 x_2 + x_1 x_2^2$ and write it as a column vector.

3.1 Basic Rules of Partial Differentiation

1. Product Rule:
$$\frac{\partial}{\partial \mathbf{x}}(f(\mathbf{x})g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}}g(\mathbf{x}) + f(\mathbf{x})\frac{\partial g}{\partial \mathbf{x}}$$
2. Sum Rule:
$$\frac{\partial}{\partial \mathbf{x}}(f(\mathbf{x}) + g(\mathbf{x})) = \frac{\partial f}{\partial \mathbf{x}} + \frac{\partial g}{\partial \mathbf{x}}$$
3. Chain Rule:
$$\frac{\partial}{\partial \mathbf{x}}(g \circ f)(\mathbf{x}) = \frac{\partial g}{\partial f}\frac{\partial f}{\partial \mathbf{x}}$$

Consider a function f of two variables x_1, x_2. Furthermore, $x_1(t)$ and $x_2(t)$ are themselves function of t. To compare the gradient of f with respect to t, we need to apply the chain rule for multivariate functions as

$$\frac{df}{dt} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial t} \end{bmatrix} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial t}$$

where d denotes the gradient and ∂ denotes partial derivative.
Exercise: Consider $f(x_1, x_2) = x_1^2 + 2x_2$, where $x_1 = \sin t$ and $x_2 = \cos t$, find df/dt.

If $f(x_1, x_2)$ is a function of x_1 and x_2, where $x_1(s,t)$ and $x_2(s,t)$ are themselves functions of two variables s and t, the chain rule yields the partial derivatives

$$\frac{df}{ds} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial s} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial s}$$
$$\frac{df}{dt} = \frac{\partial f}{\partial x_1}\frac{\partial x_1}{\partial t} + \frac{\partial f}{\partial x_2}\frac{\partial x_2}{\partial t}$$

4 Gradients of vector-valued functions

For a function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ and a vector $\mathbf{x} = [x_1, ..., x_n]^T \in \mathbb{R}^n$, the corresponding vector of function values is given as

$$\mathbf{f}(x) = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix} \in \mathbb{R}^m$$

Therefore, the gradient of $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ with respect to $\mathbf{x} = [x_1, ..., x_n]^T \in \mathbb{R}^n$ is

$$\frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix}$$

Definition: The collection of all first-order partial derivatives of a vector-values function is called the **Jacobian**. The Jacobian **J** is an $m \times n$ matrix, which we defined and arrange as follows:

$$J = \nabla_x \mathbf{f} = \frac{d\mathbf{f}(\mathbf{x})}{d\mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix}$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Exercise: Consider the function g: $h(t) = (f \circ g)(t)$ with

$$f(x_1, x_2) = e^{x_1 x_2^2}$$

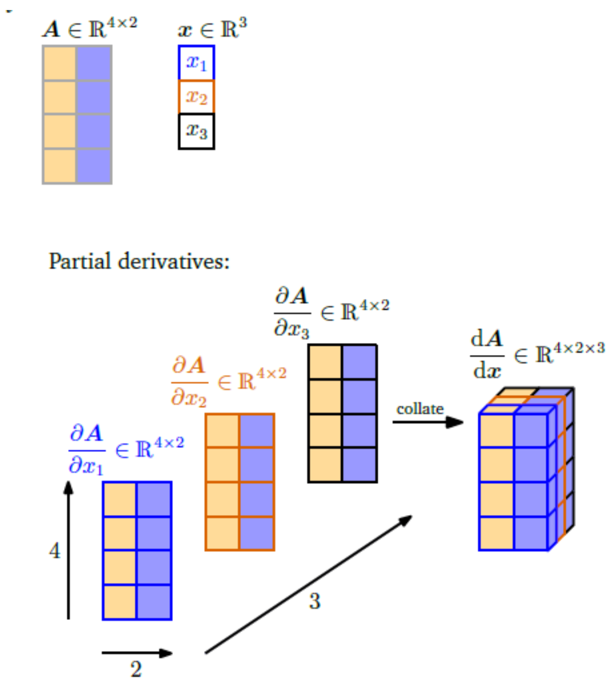
$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = g(t) = \begin{bmatrix} t\cos t \\ t\sin t \end{bmatrix}$$

Compute the gradient of h with respect to t.

$$\frac{dh}{dt} = \begin{bmatrix} \frac{\partial f}{\partial x_1} & \frac{\partial f}{\partial x_2} \end{bmatrix} \begin{bmatrix} \frac{\partial x_1}{\partial t} \\ \frac{\partial x_2}{\partial t} \end{bmatrix}$$

5 Gradients of matrices

We will encounter situations where we need to take gradients of matrices with respect to vectors, which results in a multidimentional tensor.


$\boldsymbol{A} \in \mathbb{R}^{4\times 2}$
$\boldsymbol{x} \in \mathbb{R}^{3}$
x_1
x_2
x_3
Partial derivatives:
$\frac{\partial \boldsymbol{A}}{\partial x_1} \in \mathbb{R}^{4\times 2}$
$\frac{\partial \boldsymbol{A}}{\partial x_2} \in \mathbb{R}^{4\times 2}$
$\frac{\partial \boldsymbol{A}}{\partial x_3} \in \mathbb{R}^{4\times 2}$
collate
$\frac{\mathrm{d}\boldsymbol{A}}{\mathrm{d}\boldsymbol{x}} \in \mathbb{R}^{4\times 2\times 3}$
4
2
3

Exercise: Find the derivative of

$$A = \begin{bmatrix} x_1 + x_2 & x_1x_2 & x_1x_2x_3 \\ x_1 & x_1 + x_3 & x_2^2 \\ x_2 & x_1x_3^2 & x_3^2 \end{bmatrix}$$

respect to $\mathbf{x} = [x_1 \quad x_2 \quad x_3]$.

6 Backpropagation and automatic differentiation

In many machine learning applications, we find good model parameters by performing gradient descent, which relies on the fact that we can compute the gradient of a learning objective with respect to the parameters of the model. Usually compute the gradient of the objective function and write out the result in the explicit way is often impractical since it often results in a very long expression. In other words, the implementation of the gradient could be significantly more expensive than computing the function. For training deep neural network models, the **backpropagation** algorithm is an efficient way to compute the gradient of an error function with respect to the parameters of the model.

6.1 Backpropagation

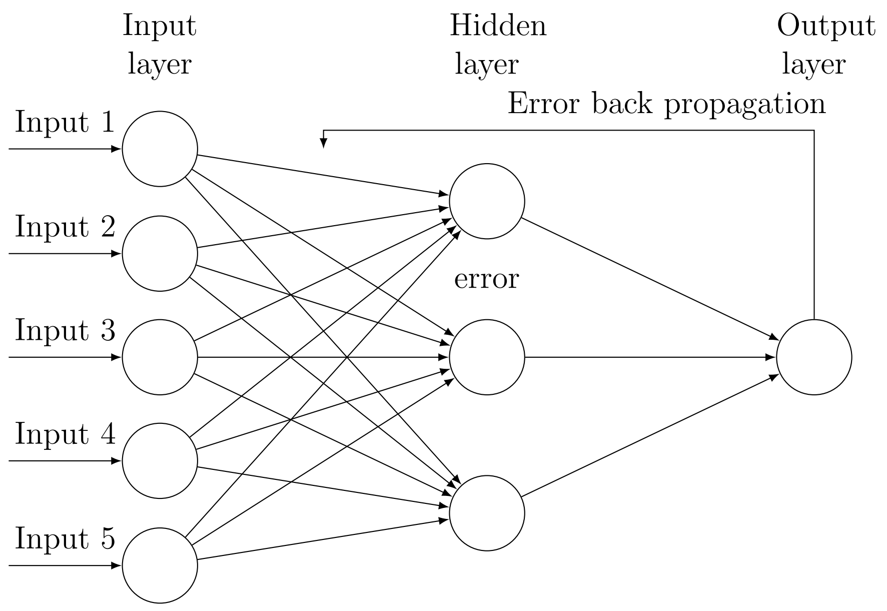
The objective of backpropagation is to change the weights for the neurons, in order to bring the error function to a minimum.

Steps:

1. Initialize weights for the parameters we want to train
2. Forward propagate through the network to get the output values
3. Define the error or cost function and its first derivatives
4. Backpropagate through the network to determine the error derivatives
5. Update the parameter estimates using the error derivative and the current value

Algorithm 1 Backpropagation learning algorithm

```
for d in data do
    FORWARDS PASS
        Starting from the input layer, to do a forward pass trough the network,
        computing the activities of the neurons at each layer.
    BACKWARDS PASS
        Compute the derivatives of the error function with respect to the
        output layer activities
        for layer in layers do
            Compute the derivatives of the error function with respect to the
            inputs of the upper layer neurons
            Compute the derivatives of the error function with respect to the
            weights between the outer layer and the layer below
            Compute the derivatives of the error function with respect to the
            activities of the layer below
        end for
        Updates the weights.
end for
```



Backpropagation Visualization: `http://www.emergentmind.com/neural-network`

6.2 Automatic differentiation

It turns out that backpropagation is a special case of a general technique in numerical analysis called automatic differentiation. We can think of automatic differentation as a set of techniques to numerically evaluate the exact gradient of a function by working with intermediate variables and applying the chain rule.

Example: Consider the function

$$f(x) = \sqrt{x^2 + e^{x^2}} + \cos(x^2 + e^{x^2})$$

Find the gradient of $f(x)$.

Idea: Use intermediate variables, let

$$a = x^2 \qquad \frac{\partial a}{\partial x} = 2x$$

$$\begin{aligned}
b &= e^a & \frac{\partial b}{\partial a} &= e^a \\
c &= a + b & \frac{\partial c}{\partial a} &= 1 = \frac{\partial c}{\partial b} \\
d &= \sqrt{c} & \frac{\partial d}{\partial c} &= \frac{1}{2\sqrt{c}} \\
e &= \cos c & \frac{\partial e}{\partial c} &= -\sin c \\
f &= d + e & \frac{\partial f}{\partial d} &= 1 = \frac{\partial f}{\partial c}
\end{aligned}$$

We can compute $\frac{\partial f}{\partial x}$ by working backward from the output and obtain

$$\begin{aligned}
\frac{\partial f}{\partial c} &= \frac{\partial f}{\partial d}\frac{\partial d}{\partial c} + \frac{\partial f}{\partial e}\frac{\partial e}{\partial c} \\
\frac{\partial f}{\partial b} &= \frac{\partial f}{\partial c}\frac{\partial c}{\partial b} \\
\frac{\partial f}{\partial a} &= \frac{\partial f}{\partial b}\frac{\partial b}{\partial a} + \frac{\partial f}{\partial c}\frac{\partial c}{\partial a} \\
\frac{\partial f}{\partial x} &= \frac{\partial f}{\partial a}\frac{\partial a}{\partial x}
\end{aligned}$$

7 Linearization and multivariate Taylor series

7.1 Linearization

In mathematics, linearization is finding the linear approximation to a function at a given point. The linear approximation of a function is the first order

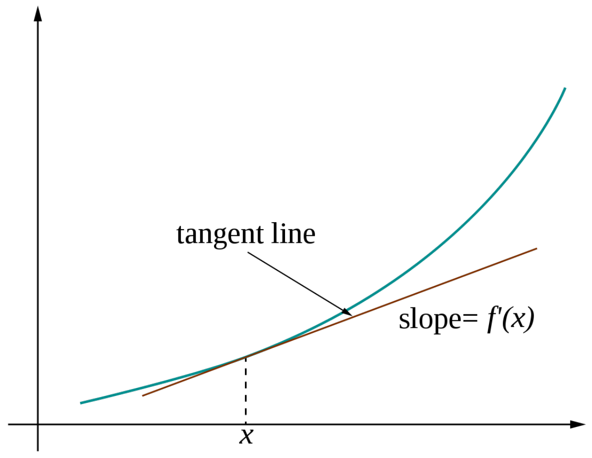


Figure 1: An approximation of $f(x)$ at $(x, f(x))$

Taylor expansion around the point of interest. If f only has one variable, the linearization function at $x = a$ is:

$$f(x) = f(a) + f'(a)(x - a)$$

Example: To find $\sqrt{4.001}$, we can use the fact that $\sqrt{4} = 2$. The linearization of $f(x) = \sqrt{x}$ at $x = a$ is $y = \sqrt{a} + \frac{1}{2\sqrt{a}}(x - a)$. We have $y = 2 + \frac{x-4}{4}$. Then we substitute $x = 4.001$, we have $\sqrt{4.001} = 2.00025$. The true value is close to 2.00024998, so the linearization approximation has a relative error of less than 1 million of a percent.

In general, if f is a vector function, the local linear approximation of f around $\mathbf{x_0}$:

$$f(\mathbf{x}) \approx f(\mathbf{x_0}) + (\nabla_x f)(\mathbf{x_0})(\mathbf{x} - \mathbf{x_0})$$

where $(\nabla_x f)(\mathbf{x_0})$ is the gradient of f with respect to $\mathbf{x}$, evaluate at $\mathbf{x_0}$.

7.2 Taylor Series

In mathematics, the Taylor series of a function is an infinite sum of terms that are expressed in terms of the function's derivatives at a single point. For most common functions, the function and the sum of its Taylor series are equal near this point. Taylor's series are named after Brook Taylor who introduced them in 1715. If zero is the point where the derivatives are considered, a Taylor series is also called a Maclaurin series, after Colin Maclaurin, who made extensive use of this special case of Taylor series in the 18th century.

The Taylor Series of a real or complex-valued uni-variable function $f(x)$ that is infinitely differentiable at point a is the power series

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-2)^2 + \cdots$$

In general, multivariate Taylor Series can be written as

$$f(\mathbf{x}) = \sum_{n=0}^{\infty} \frac{D^n_{\mathbf{x}} f(\mathbf{x_0})}{n!}(\mathbf{x} - \mathbf{x_0})^k$$

Example: Consider the function $f(x, y) = x^2 + 2xy + y^2$. Find the Taylor Series expansion of f at $(x_0, y_0) = (1, 2)$.