# Statistical Computation

Derek Li

## Contents

# 1 Basics

## 1.1 Floating Point Representation

**Definition 1.1.** A ***floating point number*** is represented by three components: $(S, F, E)$ where $S$ is the sign of the number ($\pm 1$), $F$ is a fraction (lying between 0 and 1), $E$ is an exponent. $S, F, E$ are all represented as binary digits (bits). The ***floating point representation*** of $x$, $\mathrm{fl}(x)$ is

$$\mathrm{fl}(x) = S \times F \times 2^E$$

**Note.** $x$ and $\mathrm{fl}(x)$ need not be the same, since $\mathrm{fl}(x)$ is a binary approximation to $x$, and there are only a finite number of floating point numbers.

### 1.1.1 Round-Off Error

Mathematical operations introduce further approximation errors

$$f(\mathrm{fl}(x)) = f(x + \varepsilon) \approx f(x) + \varepsilon f'(x)$$

and the goal is to make the round-off error $|f(x) - \mathrm{fl}(f(\mathrm{fl}(x)))|$ as small as possible.

### 1.1.2 Machine Epsilon and Other Constants

For a given real number $x$, we have

$$|\mathrm{fl}(x) - x| \leqslant U|x| \text{ or } \mathrm{fl}(x) = x(1 + u), |u| \leqslant U$$

where $U$ is ***machine epsilon*** or ***machine unit***. $U$ is machine dependent but very small. In R, $U = 2^{-52} = 2.220 \times 10^{-16}$.

Other machine dependent constants include:

1. The minimum and maximum positive floating point numbers: $x_{\min} = 2^{-1022} = 2.225 \times 10^{-308}$ and $x_{\max} = 2^{1024} - 1 = 1.798 \times 10^{308}$.
2. The maximum integer: $2147383647 = 2^{31} - 1$.

### 1.1.3 Overflow and Underflow Error

**Definition 1.2.** If the result of a floating point operation exceeds $x_{\max}$, then the value returned is `Inf`.

**Note.** `Inf` indicates an ***overflow error***.

**Definition 1.3.** If the result of a floating point operation is undefined then `NaN` is returned.

**Definition 1.4.** An ***underflow error*** occurs when the result of a floating point calculation is smaller (in absolute value) than $x_{\min}$.

**Note.** There are two possible outcomes (for underflow error): an error is reported or an exact 0 is returned. The latter outcome may cause problems in subsequent computations (e.g., division by 0).

**Note.** There are some ways to avoid overflow and underflow errors:

1. Use logarithmic scale: Changes multiplication/division into addition/subtraction, e.g., `lgamma`, `lfactorial`, `lchoose`.
2. Use series expansions (e.g., Taylor series).

**Example 1.1.** For $x$ close to 0, $\dfrac{\exp(x)-1}{x} \approx 1$. Naive computation of $\dfrac{\exp(x)-1}{x}$ is problematic for $x$ close to 0 due to possible round-off and underflow errors:

$$\frac{\mathrm{fl}(\exp(x)-1)}{\mathrm{fl}(x)} \neq \frac{\exp(x)-1}{x}$$

We solve the problem by using a series approximation, for $|x| \leqslant \varepsilon$,

$$\frac{\exp(x)-1}{x} = \frac{x + x^2/2 + x^3/6 + \cdots}{x} = 1 + \frac{x}{2} + \frac{x^2}{6} + \cdots$$

### 1.1.4 Catastrophic Cancellation

Suppose $z_1 = g_1(x_1, \cdots, x_n)$ and $z_2 = g_2(x_1, \cdots, x_n)$. We want to compute $y = z_1 - z_2$. What we actually compute is

$$y^* = \mathrm{fl}(\mathrm{fl}(z_1) - \mathrm{fl}(z_2))$$

where $\mathrm{fl}(z_1) = z_1(1+u_1)$ and $\mathrm{fl}(z_2) = z_2(1+u_2)$. We have

$$\mathrm{fl}(z_1) - \mathrm{fl}(z_2) = \underbrace{z_1 - z_2}_{y} + \underbrace{z_1 u_1 - z_2 u_2}_{\text{error}}$$

If $z_1$ and $z_2$ are large but $y = z_1 - z_2$ is small then the magnitude of the error may be larger than the magnitude of $y$ - ***catastrophic cancellation***.