Numerical Analysis.

- Doing math with computers
- Finding the best algorithms
  methods to solve problems
- Dealing with error.

Errors in quantities used for mathematical calculations.

If x is an exact value we will sometimes label an approximation to x as $\bar{x}$.
($\bar{x}$ is sometimes called an approximate number)

Example: $\frac{1}{3} = 0.\overline{3} = x$

On a device with 8 decimal places of

accuracy, it might render $\frac{1}{3}$

as $\bar{x} = 0.33333333$

The difference $e_x = \underbrace{x}_{\text{exact}} - \underbrace{\bar{x}}_{\text{approximate.}}$ is the error in $\bar{x}$.

$\Rightarrow x = \bar{x} + e_x$

$\Rightarrow \bar{x} = x - e_x.$

If we don't know what e_x is exactly, it's nice to know a bound on e_x; we'll label that $E_x \geq 0$, where $|e_x| \leq E_x$.

<u>Our example</u>: $x = \frac{1}{3} = 0.\overline{3}$, $\bar{x} = 0.33333333$.

Then $e_x = x - \bar{x}$

$= \frac{1}{3} - 0.33333333$

$= 0.00000000\overline{3}$

<u>Note:</u> If we know that our computer calculates fractions accurate to within

$.0000001$, $E_x = 0.0000001$,

and for any fraction input x,
the computer will estimate it as $\bar{x}$, where
$e_x = x - \bar{x}$ satisfies $|e_x| \leq E_x = 0.0000001$

The relative error in $\bar{x}$ is

$$\varepsilon_x = \varepsilon_{\bar{x}} = \frac{e_x}{\bar{x}},$$

In our example, $x = 0.\bar{3}$, $\bar{x} = 0.33333333$
$e_x = 0.000000000 3\bar{3}$

$$\Rightarrow \quad \varepsilon_x = \frac{e_x}{\bar{x}} = \frac{0.00000000 3\bar{3}}{0.33333333}$$

$$\varepsilon_x \approx 10^{-8}.$$

Often we will want an estimate on
the relative error.

Computer estimations of numbers.

1. Rounding techniques,
Given x, create a "best" $\bar{x}$.

example – truncation.
Change 13.9726 to only 3 decimal places beyond the decimal point.

$\rightarrow 13.972$ truncation

$\rightarrow 13.973$ rounding (standard)

For any input x,
Changing x so it has only 3 decimal places beyond the decimal point.

$\Rightarrow E_x = 0.001$ for truncation.

$\Rightarrow E_x = 0.0005$ for standard rounding

Computers store numbers

• Main points: numbers may be displayed as decimals, but they are actually stored in the computer as binary numbers (base-2).

Examples: (I'll use a subscript 2 when using binary format.)

$$5 = 2^2 + 2^0 = 101_2$$

By analogy:

10^2 spot, 10^1 spot, 10^0 spot.

Decimals:

$$342 = 3\cdot 10^2 + 4\cdot 10^1 + 2\cdot 10^0$$

$$3.8 = 3\cdot 10^0 + 8\cdot 10^{-1}$$

Binary #s

$$10111_2 = 1\cdot 2^4 + 0\cdot 2^3 + 1\cdot 2^2 + 1\cdot 2^1 + 1\cdot 2^0$$

$$= 16 + 4 + 2 + 1 = 23$$

$$10.111_2 = 1\cdot 2^1 + 0\cdot 2^0 + 1\cdot 2^{-1} + 1\cdot 2^{-2} + 1\cdot 2^{-3}$$

$$= \frac{10111_2}{2^3} = \frac{23}{8}$$

called the "radix point" ← in base 10, it's the decimal point.

Examples (1) What is 342 in base 2?

Plan/algorithm: find largest power of 2 that is < 342, subtract it, repeat.

$$
\begin{array}{r l}
342 & \leftarrow 2^8 = 256 \\
\underline{256} & \\
86 & \leftarrow 2^6 = 64 \\
\underline{64} & \\
22 & \leftarrow 2^4 \\
\underline{16} & \\
6 & \leftarrow 2^2 \\
\underline{4} & \\
2 & \leftarrow 2^1 \\
\underline{2} &
\end{array}
$$

In summary,

$$342 = 2^8 + 2^6 + 2^4 + 2^2 + 2^1$$

$$= 101010110_2$$

Example: What is 3.8 in binary?

$$
\begin{array}{r l}
3.8 & \leftarrow 2^1 \\
\underline{2} & \\
1.8 & \leftarrow 2^0 \\
\underline{1} & \\
.8 & \\
\underline{.5} & \leftarrow 2^{-1} \\
.3 & \\
\underline{.25} & \leftarrow 2^{-2}
\end{array}
\qquad
\begin{array}{r l}
\underline{.05} & \\
\underline{.03125} & \leftarrow 2^{-5} \\
.01875 & \leftarrow \text{Aaahh.}
\end{array}
$$

$$\Rightarrow \boxed{3.8 = 11.11001\ldots_2}$$

fram sagemath, it looks like

$$3.8 = 11.\overline{1100}_2$$