16. Algorithm stability

- cancellation
- numerical stability
Example

two expressions for the same function

\[ f(x) = \frac{1 - \cos^2 x}{x^2} \]
\[ g(x) = \frac{\sin^2 x}{x^2} \]

- results of \( \cos x \) and \( \sin x \) were rounded to 10 significant digits
- other calculations are exact
- plot shows function at 100 equally spaced points between \(-0.01\) and \(0.01\)
Evaluation of $f$

evaluate $f(x)$ at $x = 5 \cdot 10^{-5}$

- calculate $\cos x$ and round result to 10 digits

\[
\cos x = 0.99999999875000\ldots \\
\sim 0.9999999988
\]

- evaluate $f(x) = (1 - \cos(x)^2)/x^2$ using rounded value of $\cos x$

\[
\frac{1 - (0.9999999988)^2}{(5 \cdot 10^{-5})^2} = 0.9599\ldots
\]

has only one correct significant digit (correct value is 0.9999\ldots)
Evaluation of $g$

evaluate $g(x)$ at $x = 5 \cdot 10^{-5}$

- calculate $\sin x$ and round result to 10 digits
  \[
  \sin x = 0.499999999791667 \ldots \cdot 10^{-5}
  \sim 0.4999999998 \cdot 10^{-5}
  \]

- evaluate $f(x) = \sin(x)^2/x^2$ using rounded value of $\cos x$
  \[
  \frac{(\sin x)^2}{x^2} \approx \frac{(0.4999999998 \cdot 10^{-5})^2}{(5 \cdot 10^{-5})^2} = 0.9999 \ldots
  \]
  has about ten correct significant digits

Conclusion: $f$ and $g$ are equivalent mathematically, but not numerically
Cancellation

\[ \hat{a} = a(1 + \Delta a), \quad \hat{b} = b(1 + \Delta b) \]

- \(a, b\): exact values
- \(\hat{a}, \hat{b}\): approximations with unknown relative errors \(\Delta a, \Delta b\)
- relative error in \(\hat{x} = \hat{a} - \hat{b} = (a - b) + (a\Delta a - b\Delta b)\) is

\[
\frac{|\hat{x} - x|}{|x|} = \frac{|a\Delta a - b\Delta b|}{|a - b|}
\]

if \(a \approx b\), small \(\Delta a\) and \(\Delta b\) can lead to very large relative errors in \(x\)

this is called **cancellation**; cancellation occurs when:

- we subtract two numbers that are almost equal
- one or both numbers are subject to error
Cancellation occurs in the example when we evaluate the numerator of

\[ f(x) = \frac{1 - (\cos x)^2}{x^2} \]

- \(1 \approx (\cos x)^2\) when \(x\) is small
- there is a rounding error in \(\cos x\)
Numerical stability

refers to the accuracy of an algorithm in the presence of rounding errors

• an algorithm is *unstable* if rounding errors cause large errors in the result
• rigorous definition depends on what ‘accurate’ and ‘large error’ mean
• instability is often, but not always, caused by cancellation

**Examples** from earlier lectures

• solving linear equations by LU factorization without pivoting
• Cholesky factorization method for least squares
Roots of a quadratic equation

\[ ax^2 + bx + c = 0 \quad (a \neq 0) \]

**Algorithm 1:** use the formulas

\[ x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a} \]

unstable if \( b^2 \gg |4ac| \)

- if \( b^2 \gg |4ac| \) and \( b \leq 0 \), cancellation occurs in \( x_2 \) \((-b \approx \sqrt{b^2 - 4ac})\)
- if \( b^2 \gg |4ac| \) and \( b \geq 0 \), cancellation occurs in \( x_1 \) \((b \approx \sqrt{b^2 - 4ac})\)
- in both cases \( b \) may be exact, but the squareroot introduces small errors
Roots of a quadratic equation

\[ ax^2 + bx + c = 0 \quad (a \neq 0) \]

**Algorithm 2:** use fact that roots \(x_1, x_2\) satisfy \(x_1x_2 = c/a\)

- if \(b \leq 0\), calculate
  
  \[
  x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}, \quad x_2 = \frac{c}{ax_1}
  \]

- if \(b > 0\), calculate
  
  \[
  x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}, \quad x_1 = \frac{c}{ax_2}
  \]

no cancellation when \(b^2 \gg |4ac|\)
Exercises

- `chop(x,n)` rounds $x$ to $n$ significant decimal digits
- for example `chop(pi,4)` returns 3.14200000000000

**Exercise 1:** cancellation occurs in $\frac{1 - \cos x}{\sin x}$ when $x \approx 0$

```matlab
>> x = 1e-2;
>> (1 - chop(cos(x), 4)) / chop(sin(x), 4)
ans =
    0
```

(exact value is about 0.005)

give a stable alternative method
Exercise 2: Euler proved that \( \sum_{k=1}^{\infty} k^{-2} = \pi^2/6 = 1.644934 \ldots \)

the sum of the first 3000 terms is

\[
\sum_{k=1}^{3000} k^{-2} = 1.6446
\]

we compute this sum rounding all intermediate results to 4 digits:

```matlab
>> sum = 0;
>> for k = 1:3000
    sum = chop(sum + 1/k^2, 4);
end
>> sum
sum =
    1.6240
```

• result has only two correct digits
• not caused by cancellation (there are no subtractions)

explain and propose a better method
Exercise 3: the number $e = 2.7182818 \cdots$ can be defined as

$$e = \lim_{n \to \infty} (1 + 1/n)^n$$

this suggests an algorithm for calculating $e$: take a large $n$ and evaluate

$$\hat{e} = (1 + 1/n)^n$$

results:

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\hat{e}$</th>
<th>Number of correct digits</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^4$</td>
<td>2.718145926</td>
<td>4</td>
</tr>
<tr>
<td>$10^8$</td>
<td>2.718281798</td>
<td>7</td>
</tr>
<tr>
<td>$10^{12}$</td>
<td>2.718523496</td>
<td>4</td>
</tr>
<tr>
<td>$10^{16}$</td>
<td>1.0000000000</td>
<td>0</td>
</tr>
</tbody>
</table>

explain
Exercise 4: on page 2.11 we showed that for an $n$-vector $x$,

$$\text{std}(x)^2 = \frac{1}{n} \|x - \text{avg}(x)1\|^2 = \frac{1}{n} \left( \|x\|^2 - \frac{(1^T x)^2}{n} \right)$$

we evaluate the second expression for $n = 10$ and $x = (1002, 1000, 1003, 1001, 1002, 1002, 1001, 1004, 1002, 1001)$

```matlab
>> sum1 = 0.0; sum2 = 0.0;
>> for i = 1:n
    sum1 = chop( sum1 + x(i), 6 );
    sum2 = chop( sum2 + x(i)^2, 6 );
>> end
>> s = chop( ( sum2 - sum1^2 / n ) / n, 6)
```

```
s =
  -3.2400
```

a negative number! explain and suggest a better method