Lecture 5 Review

- Basic rule for numerical integration $I = \sum f(x_i) w_i ; i = 1, N$

- Trapezoidal rule: $w_i = \{ h/2, h, h, \ldots h, h, h/2\}$,
  $h = (b - a)/N, N = \text{odd}$.

- Simpson’s rule: $w_i = \{ h/3, 4h/3, 2h/3, 4h/3, 2h/3, \ldots 2h/3, 4h/3, h/3\}$,
  $h = (b - a)/N, N = \text{odd}$.

http://www.physics.smu.edu/devel/coan/3340/simpson.cc
http://www.cplusplus.com/doc/tutorial/control.html
http://www.cplusplus.com/reference/clibrary/cmath/
ERROR ESTIMATION IN NUMERICAL INTEGRATION

WE NEED TO ESTIMATE OUR ERRORS IN OUR NI FORMULAE.

Q: HOW WELL DO WE DO WHEN WE USE MORE POINTS (i.e., INCREASE N)?

USE STD RESULTS FROM NUMERICAL METHODS

SEE www.nrbook.com/abramowitz-and-stegun/page-1805.htm

\[
\int_{x_0}^{x_m} f(x) \, dx = h \left[ \frac{f_0}{2} + f_1 + \ldots + f_{m-1} + \frac{f_m}{2} \right]
\]

\[- \frac{m \cdot h^3}{12} f^{(3)}(\xi) \]  

(ABS NOTATION)  \( w/ \) \( x_0 < \xi < x_m \)
**Trapezoidal Rule:**

\[
\text{ERROR} = \frac{m h^3}{12} f''(\xi) \quad \text{unknown}
\]

\[m = N\]

\[h = \frac{1}{N}\]

\[\Rightarrow \text{ERROR} = O\left(\frac{1}{N^2}\right)\quad \text{TRAPEZOIDAL RULE}\]

**Note:** A 8S ~ "extended" TR

**Simpson's Rule:**

\[
\int_{x_0}^{x_{2n}} f(x) \, dx = \frac{h}{3} \left[ f_0 + 4 (f_1 + f_3 + \ldots + f_{2n-1}) - \frac{h^5}{90} f(4) (\xi) \right]
\]

\[m = N/2, \ h = \frac{1}{N}\]
SIMPSON'S RULE ERROR ESTIMATE

\[ \text{ERROR} = O\left(\frac{1}{N^4}\right) \]

NB: A8S "EXTENDED" SR

http://www.nrbook.com/abramowitz_and_stegun/page_885.htm
MONTE CARLO INTEGRATION

THROW DARTS AT DART BOARD

\[ y_1 \]

Probablity of landing in circle = proportional to area of circle

But area = integral under curve

Consider upper quadrant \( 0 \leq x \leq 1, 0 \leq y \leq 1 \)

Area in quad = \( \frac{1}{4} \int y \, dx \)
MC INTEGRATION (2)

\[ A = \int_{0}^{1} \sqrt{1-x^2} \, dx \]

"THROW A DART" MEANS PICK RANDOM NUMBER
RANDOM \( \neq \) CAN'T BE PREDICTED.

PRACTICALLY, DONE BY FUNCTION CALL.

PICK RANDOM \( x \) \( \text{ such that } 0 \leq x \leq 1 \)
PICK RANDOM \( y \) \( \text{ such that } 0 \leq y \leq 1 \)

IF \( y \leq \sqrt{1-x^2} \), THEN DART IN QUADRANT

REPEAT MANY TIMES, SAY, \( N_T \) TIMES
CALCULATE HOW MANY DARTS \( N \)

\[ A = \left( \frac{N}{N_T} \right) \times \text{ AREA OF SQUARE} \]
// computes some random numbers

#include <iostream>

using namespace std;

int main()
{
    srand((time(0))); // "seed" the random number generator
    int r = random();

    // RAND_MAX is largest random number. it is an integer. built-in.

    cout << " random number: " << (double) random()/RAND_MAX << endl;
    cout << " random number: " << random() << endl;
    cout << " random number: " << random() << endl;

    cout << "The value of RAND_MAX is " << RAND_MAX << endl;
    return 0;
}
Plotting with Gnuplot (just a peek)

We need to be able to plot. Use `gnuplot` utility.
Try it! (Start by typing `gnuplot` to enter utility.)

Usage examples:

```
gnuplot> plot sin(x)
gnuplot> plot [1:4] exp(-x) lw 2
```

```
gnuplot> plot "data_file.dat"
```

Must already exist

http://www.gnuplot.info/help.html
http://sparky.rice.edu/~hartigan/gnuplot.html

Link also available from PHYS 3340 links page

Q: plot 500 random numbers (0,1).
Help with NI errors and gnuplot

My head is exploding.

I need something to read quietly, at my own pace.

http://www.nrbook.com/abramowitz_and_stegun/page_885.htm
http://www.gnuplot.info

Link also available from PHYS 3340 links page
Summary

- Errors for trapezoidal & Simpson’s NI techniques.
- Monte Carlo (”dart throwing”) technique for integration.
- Code for random number generation.
- C++ features: if, declaring functions, random, ...
- Just a peek at plotting: gnuplot.

http://www.gnuplot.info/help.html
http://sparky.rice.edu/~hartigan/gnuplot.html

Don’t suffer in silence. Scream for help!!!