Error Propagation in a Simple Function Evaluation

Sean Rodby, Luke Snyder, Autar Kaw
University of South Florida
United States of America
kaw@eng.usf.edu
Website: http://numericalmethods.eng.usf.edu
Version: *Mathematica 6*; Last Revised: August 28, 2008

Initialization

Clearing the definitions of all symbols in the current context:

```
ClearAll[Evaluate[Context[] <> "*"]]
```

Introduction

The following worksheet illustrates how error propagates in a simple function evaluation. This is due largely to the fact that relative error calculations are based on round-off errors or/and measurement errors that exist inherently in simple devices used to take measurements. The user will enter force, length or width of the cross section, and Young's Modulus data in the *Input* section of the program, as well as the corresponding relative measurement error for each input. From here, *Mathematica* will be used to demonstrate how this error propagates. The formula used for illustration is $\text{Strain} = \frac{F}{h^2 E}$.

Section 1: Input Data

This is the only section where the user interacts with the program.

Enter inputs to calculate strain based on the formula $\text{Strain} = \frac{F}{h^2 E}$.

- Enter Force (in N)

```
Force = 72;
```

- Enter relative measurement error in percentage for force

```
RMEForce = 2.5;
```

- Enter length or width of cross section (in m):

```
h = 4 × 10⁻³;
```

- Enter relative measurement error in percentage for h

```
RMEh = 2.5;
```

- Enter Young's Modulus (in Pa)

```
E = 70 × 10⁹;
```

- Enter relative measurement error in percentage for Modulus

```
RMEYoung = 2.5;
```

This is the end of the user section. All information must be entered before proceeding to the next section. **RE-EVALUATE THE NOTEBOOK**.

Section 2: Procedure

Calculating the measurement error for each input.

```
MeasErrorForce = Force * (RMEForce / 100)
MeasErrorh = h * (RMEh / 100)
MeasErrorYoung = E * (RMEYoung / 100)

1.8

0.0001

1.75 × 10⁹
```

First we must calculate the strain given by the equation $\varepsilon = \frac{F}{h^2 E}$.

```
        Force
strain = ──────
         h² E

   9
─────────
 140 000
```

Using the formula for maximum possible error to derive the maximum possible error in the measured strain. Note that different variable names are used because the previous variable names represented actual numbers. Meaning that if the partial differential is taken, in all cases it will be zero. The *With* command is then used to input the values given by the user into the partial differential equation.

```
strain1 = F / (h1² E1);
partialForce := Abs[D[strain1, F]];
partialwrtForce = With[{E1 = E, h1 = h}, Evaluate[partialForce] ];
partialh := Abs[D[strain1, h1]];
partialwrth = With[{h1 = h, F = Force, E1 = E}, Evaluate[partialh] ];
partialYoung := Abs[D[strain1, E1]];
partialwrtYoung = With[{F = Force, h1 = h, E1 = E}, Evaluate[partialYoung] ];
```

Calculating the range of value for each measured quantity based in the RME percentages given.

```
RangeForce = ( RMEForce / 100 ) Force;
Rangeh = ( RMEh / 100 ) h;
RangeYoung = ( RMEYoung / 100 ) E;
```

Calculating the maximum possible error by adding the absolute value of the partial differential equations multiplied by their respective range of variation in the measured value. Then calculating the individual error contributions due to Force, Area, and the

length or width dimension respectively.

```
(*With respect to force*)
ErrorForce = partialwrtForce * RangeForce
```

1.60714×10^{-6}

```
(*With respect to h*)
Errorh = partialwrth * Rangeh
```

3.21429×10^{-6}

```
(*With respect to Young's Modulus*)
ErrorYoung = partialwrtYoung * RangeYoung
```

1.60714×10^{-6}

```
(*Calculating total error by adding all the
 individual contributions given by the user defined inputs.*)
TotError = ErrorForce + Errorh + ErrorYoung
```

6.42857×10^{-6}

```
(*Calculating the percent contribution of force error to the total error.*)
```
$$RelForceError = \frac{ErrorForce * 100}{TotError}$$

25.

```
(*Calculating the percent contribution of h error to the total error.*)
```
$$RelhError = \frac{Errorh * 100}{TotError}$$

50.

```
(*Calculating the percent contribution of Young's Modulus error to the total error.*)
```
$$RelYoungError = \frac{ErrorYoung * 100}{TotError}$$

25.

```
(*Calculating the effective range that the axial
 strain could be within using the total error calculation.*)
Lowstrain = strain - TotError
Highstrain = strain + TotError
```

0.0000578571

0.0000707143

```
(*Calculating the maximum percent relative measurement error for strain.*)
```
$$RMEstrain = \frac{100 * TotError}{strain}$$

10.

Conclusion

If a calculation is made with numbers that are not exact, then the calculation itself will have an error. Since the final results of an experiment are not usually directly measured but are some function of one or more of the measured quantities, it is important to understand and utilize propagation of error concepts to better interpret and represent experimental results.

References

Propagation of Errors.
See: http://numericalmethods.eng.usf.edu/nbm/gen/01aae/nbm_gen_aae_txt_propagationoferrors.pdf