

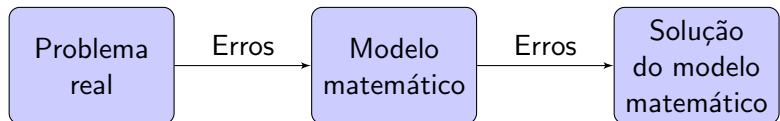
Erros

Evandro Alves Nakajima

Universidade Tecnológica Federal do Paraná
Campus Santa Helena

2017





A precisão dos dados de entrada bem como as operações numéricas influenciam nos resultados.

Erros na fase de modelagem: São erros provenientes de simplificações que muitas vezes são necessárias para que um fenômeno real possa ser tratado pelas ferramentas matemáticas disponíveis.

Erros na fase de solução: São erros provenientes da utilização de algum equipamento, como por exemplo o computador.

Exemplo: Fase de modelagem

Considere a aproximação de π com 21 casas decimais

$$\pi = 3.141592653589793238462$$

Vamos calcular a área de uma circunferência com $1000m$ de raio.

$$\begin{aligned}\text{Área} &= 3.141592653589793238462 \times (1000m)^2 \\ &= 3141592.653589793238462m^2\end{aligned}$$

O programa excel, por exemplo, toma $\pi = 3.14159265358979$, assim

$$\text{Área} = 3.14159265358979 \times (1000m)^2 = 3141592.65358979m^2$$

Obtemos aqui um erro igual à $0.000000003238462m^2$

Duas aproximações normalmente usadas para o número π quando queremos fazer contas à mão são $\pi = 3.14$ e $\pi = 3.1415$

Para esses dois casos temos

$$\text{Área} = 3.1415 \times (1000m)^2 = 3141500m^2$$

Que nos dá um erro superior a $92m^2$, e

$$\text{Área} = 3.14 \times (1000m)^2 = 3140000m^2$$

Que nos dá um erro superior a $1592m^2$.

Exemplo: Fase de solução

Vamos executar o seguinte somatório:

$$\sum_{i=1}^{30000} 0.2$$

Sabemos que somar 30000 o número 0.2 é o mesmo que realizar a multiplicação

$$0.2 \times 30000 = 6000$$

Façamos esse procedimento utilizando o programa *DevC++* de duas formas:

```
int main(int argc, char** argv) {  
    float soma = 0;  
  
    int i;  
  
    for (i = 1; i ≤ 30000; i++)  
        soma = soma + 0.2;  
  
    printf("%.20f \n \n", soma);  
  
    system("pause");  
}
```

Aqui devemos obter o resultado 6001.15283203125 que gera um erro igual a

1.15283203125

```
int main(int argc, char** argv) {  
    double soma = 0;  
    int i;  
    for (i = 1; i ≤ 30000; i++)  
        soma = soma + 0.2;  
    printf("%.20f \n \n", soma);  
    system("pause");  
}
```

Aqui devemos obter o resultado 5999.99999999967340052593 que gera um erro igual a

0.0000000032659947407

Erro absoluto e Erro relativo

Definimos o erro absoluto como sendo

$$E_a = |x_{ex} - x_{aprox}|$$

onde x_{ex} é o valor exato da grandeza e x_{aprox} é o valor aproximado da mesma.

Definimos o erro relativo como sendo

$$E_r = \frac{|x_{ex} - x_{aprox}|}{|x_{ex}|}$$

Como normalmente não conhecemos o valor de x_{ex} é comum tomarmos um limitante superior ε para o erro.