



INSTITUTO  
TECNOLÓGICO  
DE AERONÁUTICA

Introdução à Física  
Experimental

Prof. Douglas Leite  
[www.lpp.ita.br/leite](http://www.lpp.ita.br/leite)

# Ajuste de modelos matemáticos não lineares em dados experimentais usando o Mathematica®

## Aula 2

Vídeo 1 – Introdução e atividades da aula

Vídeo 2 – Tutorial: Básico do Mathematica II

Vídeo 3 – Tutorial: Ajuste de curva no Mathematica



# Objetivos da Aula

## Objetivo Principal

- Ajustar curva de modelo matemático em dados experimentais de um Oscilador Harmônico Amortecido

## Objetivos Secundários

- Aprender o básico da linguagem do Software Mathematica® - parte numérica: listas e gráficos
- Ajustar curva teórica em dados experimentais
- Analisar qualidade do ajuste pelos resíduos

## Desafios

- Levantar os gráficos de energia potencial, energia cinética e energia total mecânica do sistema



# Atividades da aula

## Assistir aos Tutoriais

- Vídeo 2 - Básico do Mathematica II (dados numéricos, listas, importando dados, gráficos de experimentos)
- Vídeo 3 - Ajuste de curva em experimento de MHS

## Fazer Pesquisa Bibliográfica

- Energias no Movimento Harmônico Amortecido (amortecimento proporcional à velocidade) caso subcrítico

## Entregável

- Seguir roteiro do slide a seguir
- Entregar notebook no Classroom com o nome *T#\_G§§\_aula2.nb* (# = num turma, §§ = num grupo)  
(Exemplo: entregável do grupo 8 da turma 3 → *T3\_G08\_aula2.nb*)
- Não esquecer de incluir CABEÇALHO NO NOTEBOOK COM OS NOMES E E-MAIL DOS INTEGRANTES DO GRUPO



# Roteiro

## No Mathematica:

1. Importar dados do arquivo fornecido MHA\_\$\$\$.txt (\$\$ = número do grupo) e gerar seu gráfico
2. Propor modelo matemático analítico que descreve o experimento e gerar gráfico juntamente com a curva experimental (não precisa usar solução de EDO – pode usar função analítica definida diretamente:  $f[x_] = x0*\text{Sin}[...]...$ )
3. Ajustar manualmente os parâmetros do modelo (usando *Manipulate*)
4. Desenvolver rotina para minimização da soma dos quadrados dos resíduos (ajuste automático) e análise da qualidade do ajuste
5. Obter o valor da constante de mola (**k**) e da constante de amortecimento (**b**) com a respectiva incerteza (se necessário, considerar incerteza ZERO para os parâmetros ajustados, levar em consideração a incerteza da massa informada)
6. Comentar e formatar o notebook (utilizando células de texto e divisões de seções em locais apropriados) (explicar o que foi feito em cada célula, explicar os gráficos e interpretar os resultados e citar referências utilizadas)
7. Desafio: (em seção separada no final do notebook):  
Gerar gráfico com as curvas de energias potencial, cinética e mecânica, a partir da função do modelo ajustado nos dados experimentais. Discutir a relação entre as curvas e a formação de “platôs” na curva de energia mecânica total

## Pesos:

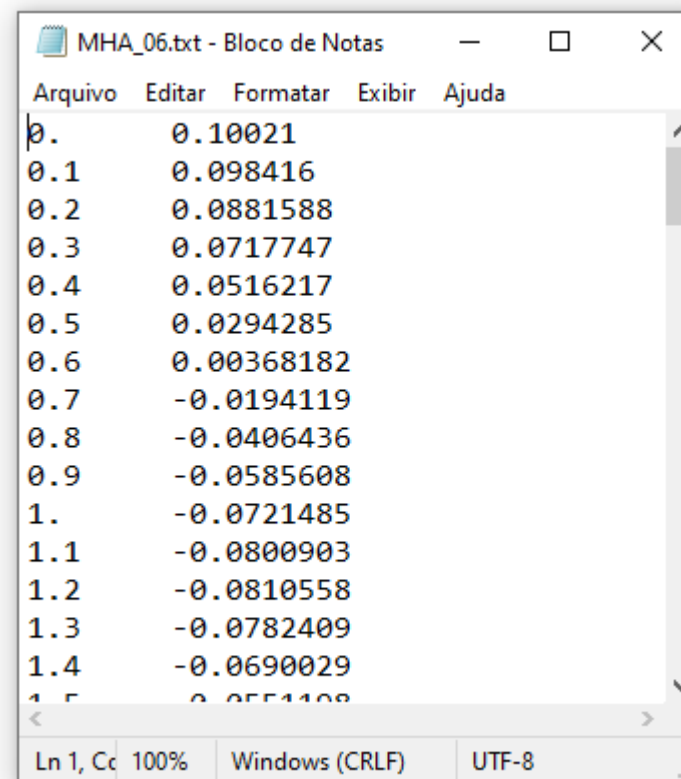
1 - 3) 2,0 pontos; 4) 4,0 pontos; 5) 2,0 pontos; 6) 2,0 pontos; 7) 2,0 pontos (adicionais)



# Dados Fornecidos

## Dados experimentais de MHA:

- Arquivo MHA\_§§.txt (§§ = número do grupo)
  - experimento massa-mola com amortecimento por fluido
  - duas colunas (tempo [s], x [m])
- MHA\_01, MHA\_05, MHA\_06  
 $m = 0.270 \pm 0.005 \text{ kg}$
- MHA\_02, MHA\_03, MHA\_04, MHA\_07, MHA\_08  
 $m = 0.0710 \pm 0.0005 \text{ kg}$
- MHA\_09, MHA\_10  
 $m = 0.872 \pm 0.015 \text{ kg}$



|     |            |
|-----|------------|
| 0.  | 0.10021    |
| 0.1 | 0.098416   |
| 0.2 | 0.0881588  |
| 0.3 | 0.0717747  |
| 0.4 | 0.0516217  |
| 0.5 | 0.0294285  |
| 0.6 | 0.00368182 |
| 0.7 | -0.0194119 |
| 0.8 | -0.0406436 |
| 0.9 | -0.0585608 |
| 1.  | -0.0721485 |
| 1.1 | -0.0800903 |
| 1.2 | -0.0810558 |
| 1.3 | -0.0782409 |
| 1.4 | -0.0690029 |
| 1.5 | -0.0551108 |

# Dicas

## Modelo e energias:

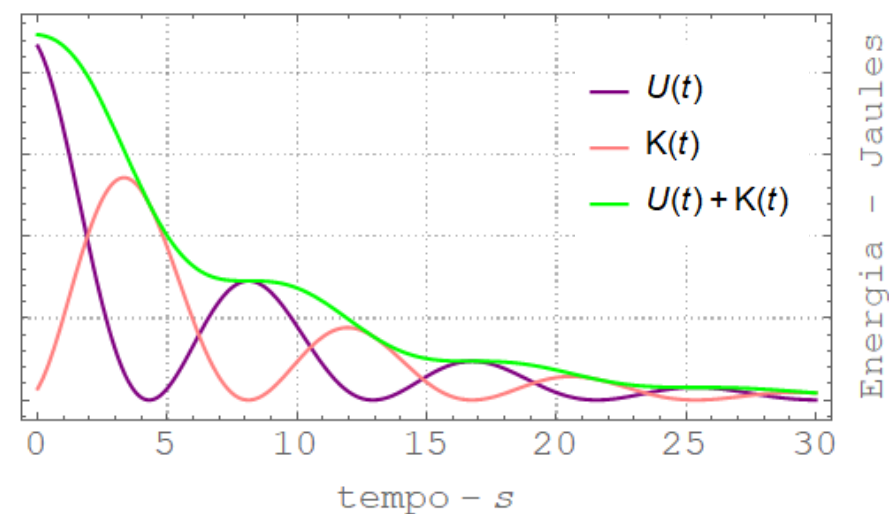
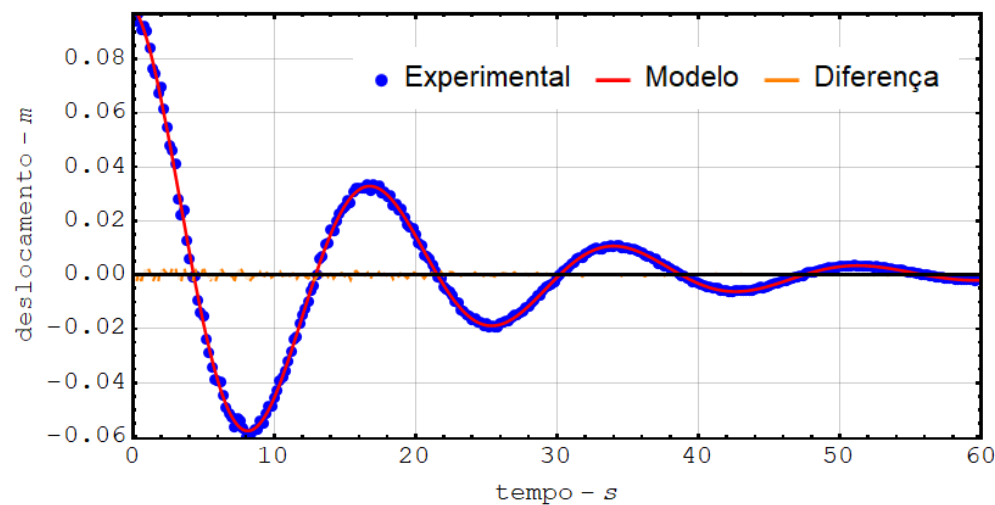
$$\frac{d^2x}{dt^2} + \gamma \frac{dx}{dt} + \omega_0^2 x = 0$$

$$\gamma = \frac{b}{m}$$

$$\omega_0 = \sqrt{\frac{k}{m}}$$

$$x(t) = Ae^{\frac{-\gamma t}{2}} \cos(\omega t + \phi)$$

$$\omega^2 = \omega_0^2 - \frac{\gamma^2}{4}$$



# Dicas

## Comando FindMinimum para ajuste automático

### FindMinimum (encontra o mínimo)

`FindMinimum[f, x]`

searches for a local minimum in  $f$ , starting from an automatically selected point.

`FindMinimum[f, {x, x0}]`

searches for a local minimum in  $f$ , starting from the point  $x = x_0$ .

`FindMinimum[f, {{x, x0}, {y, y0}, ...}]`

searches for a local minimum in a function of several variables.

`FindMinimum[{f, cons}, {{x, x0}, {y, y0}, ...}]`

searches for a local minimum subject to the constraints  $cons$ .

`FindMinimum[{f, cons}, {x, y, ...}]`

starts from a point within the region defined by the constraints.

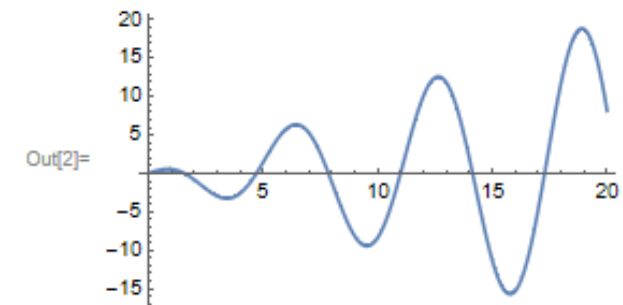
#### ▼ Basic Examples (4)

Find a local minimum, starting the search at  $x = 2$ :

```
In[1]:= FindMinimum[x Cos[x], {x, 2}]
```

```
Out[1]= {-3.28837, {x -> 3.42562}}
```

```
In[2]:= Plot[x Cos[x], {x, 0, 20}]
```



Extract the value of  $x$  at the local minimum:

```
In[3]:= x /. Last[FindMinimum[x Cos[x], {x, 2}]]
```

```
Out[3]= 3.42562
```



# Dicas

## Comando FindFit para ajuste automático

### FindFit

(encontra parâmetros de ajuste)

UPDATED IN 12

`FindFit[data, expr, pars, vars]`

finds numerical values of the parameters *pars* that make *expr* give a best fit to *data* as a function of *vars*.

`FindFit[data, {expr, cons}, pars, vars]`

finds a best fit subject to the parameter constraints *cons*.

Find a constrained fit:

```
In[1]:= model = a Cos[ $\omega$  t];
```

```
In[2]:= data = Table[{t, Cos[2.1 t]}, {t, 0, 10, .25}] + RandomReal[.1, 41];
```

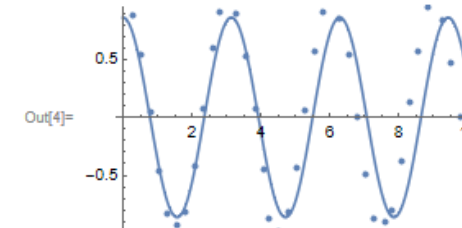
Fit to the model with positive amplitude and frequency between 1 and 2:

```
In[3]:= fit = FindFit[data, {model, {a > 0, 1 <  $\omega$  < 2}}, {a,  $\omega$ , t}
```

```
Out[3]:= {a → 0.864565,  $\omega$  → 2.}
```

Compare the fitted model to the data:

```
In[4]:= Show[Plot[model /. fit, {t, 0, 10}], ListPlot[data, PlotRange → All]]
```

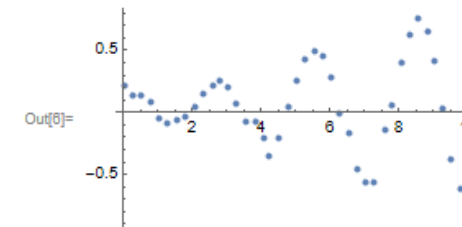


The residuals show a pattern, indicating that the frequency constraint is too strict:

```
In[5]:= residuals =
```

```
Apply[Function[{t, y}, Evaluate[{t, y - model /. fit}]], data, 1];
```

```
In[6]:= ListPlot[residuals, PlotRange → All]
```





# Dicas

## Comando FindFit para ajuste automático

### FindFit

(encontra parâmetros de ajuste)

UPDATED IN 12

`FindFit[data, expr, pars, vars]`

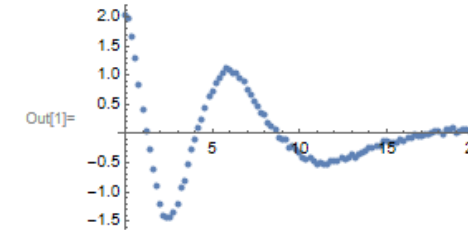
finds numerical values of the parameters *pars* that make *expr* give a best fit to *data* as a function of *vars*.

`FindFit[data, {expr, cons}, pars, vars]`

finds a best fit subject to the parameter constraints *cons*.

Find parameters in a differential equation model that give the best fit:

```
In[1]:= sol = First[x /. NDSolve[{x''[t] + .33 x'[t] + .72 x[t]^3 == 0,  
    x[0] == 2, x'[0] == 0}, x, {t, 20}]];  
times = N[Range[0, 100] / 5];  
data = Transpose[{times, sol[times] + RandomReal[.1, 101]}];  
lp = ListPlot[data, PlotRange -> All]
```



Define a model so the solution for a set of parameters ( $\gamma$ ,  $a$ ,  $b$ ) is reused for all  $x$  values:

```
In[2]:= model[\gamma_?NumberQ, a_?NumberQ, b_?NumberQ] :=  
    (model[\gamma, a, b] = First[x /. NDSolve[{x''[t] + \gamma x'[t] + a x[t] + b x[t]^3 ==  
    0, x[0] == 2, x'[0] == 0}, x, {t, 20}]]];
```

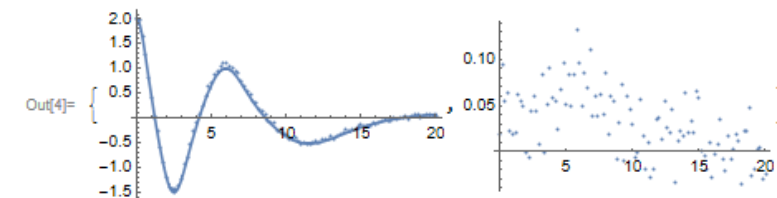
Find the fit with tolerance appropriate to the precision of the solutions of the ODE:

```
In[3]:= fit = FindFit[data, model[\gamma, a, b][x],  
    {{\gamma, .1}, {a, .1}, {b, 1}}, x, PrecisionGoal -> 4, AccuracyGoal -> 4]
```

Out[3]=  $\{\gamma \rightarrow 0.341258, a \rightarrow 0.0361403, b \rightarrow 0.712254\}$

Show the fitted model with the data and the residuals:

```
In[4]:= {Show[Plot[model[\gamma, a, b][x] /. fit, {x, 0, 20}], lp], ListPlot[  
    Transpose[{times, data[All, 2]} - (model[\gamma, a, b][times] /. fit)]]}
```



# Dicas

## Comando FindFit para ajuste automático

### FindFit

(encontra parâmetros de ajuste)

UPDATED IN 12

`FindFit[data, expr, pars, vars]`

finds numerical values of the parameters *pars* that make *expr* give a best fit to *data* as a function of *vars*.

`FindFit[data, {expr, cons}, pars, vars]`

finds a best fit subject to the parameter constraints *cons*.

#### ▼ Properties & Relations (6)

`FindFit` gives the parameter estimates:

```
In[1]:= data = Table[{i, Exp[RandomReal[{i - 1, i}]]}, {i, 10}];
```

```
In[2]:= FindFit[data, Exp[a + b x], {a, b}, x]
```

```
Out[2]:= {a → -2.72603, b → 1.24764}
```

`NonlinearModelFit` allows for extraction of additional information about the fitting:

```
In[3]:= nlm = NonlinearModelFit[data, Exp[a + b x], {a, b}, x]
```

```
Out[3]:= FittedModel[
$$e^{-2.72603 + 1.24764 x}$$
]
```

Extract the parameter estimates:

```
In[4]:= nlm["BestFitParameters"]
```

```
Out[4]:= {a → -2.72603, b → 1.24764}
```

Extract additional results and diagnostics:

```
In[5]:= nlm[{"BestFit", "FitResiduals", "ParameterTable"}]
```

```
Out[5]:= {
$$e^{-2.72603 - 1.24764 x}$$
, {1.08219, 3.12207, 16.2154, 24.9605, 104.08, 222.406, 588.695, -66.6804, -132.326, 27.8251},

|   | Estimate | Standard Error | t Statistic | P-Value                  |
|---|----------|----------------|-------------|--------------------------|
| a | -2.72603 | 0.410164       | -6.64618    | 0.000161446              |
| b | 1.24764  | 0.0413678      | 30.1597     | $1.58529 \times 10^{-9}$ |

}
```

# Dicas

## Comando Around para expressar e propagar incertezas

[ EXPERIMENTAL ]

### Around

NEW IN 12

`Around[x,  $\delta$ ]`

represents an approximate number or quantity with a value around  $x$  and an uncertainty  $\delta$ .

`Around[x, { $\delta_-$ ,  $\delta_+$ }]`

represents a number or quantity with a value around  $x$  and asymmetric uncertainties  $\delta_-$ ,  $\delta_+$ .

#### ▼ Examples (29)

##### ▼ Basic Examples (10)

Uncertain numbers of different sizes and uncertainties:

```
In[1]:= Around[1.3421, 0.012]
```

```
Out[1]= 1.342 ± 0.012
```

```
In[2]:= Around[-1.3421 × 10-7, 1.2 × 10-8]
```

```
Out[2]= (-1.34 ± 0.12) × 10-7
```

```
In[3]:= Around[RandomReal[{-1, 1}], 10-8]
```

```
Out[3]= 0.2822663 (38 ± 10)
```

Perform operations with `Around` objects:

```
In[1]:= Around[1, .1] + Around[2, .4]
```

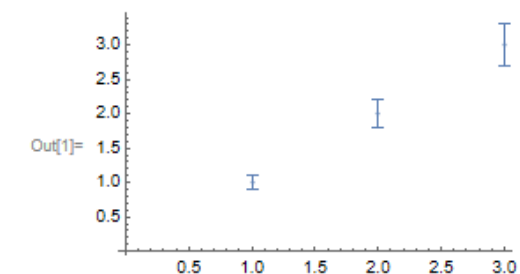
```
Out[1]= 3.0 ± 0.4
```

```
In[2]:= Exp[Around[2, .1]]
```

```
Out[2]= 7.4 ± 0.7
```

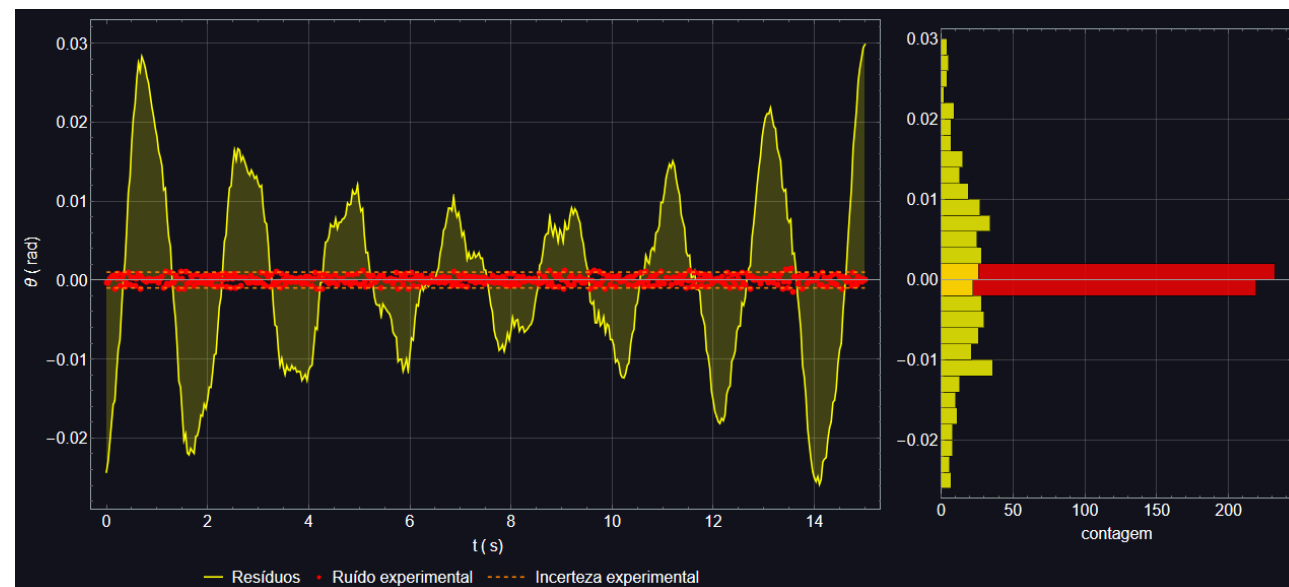
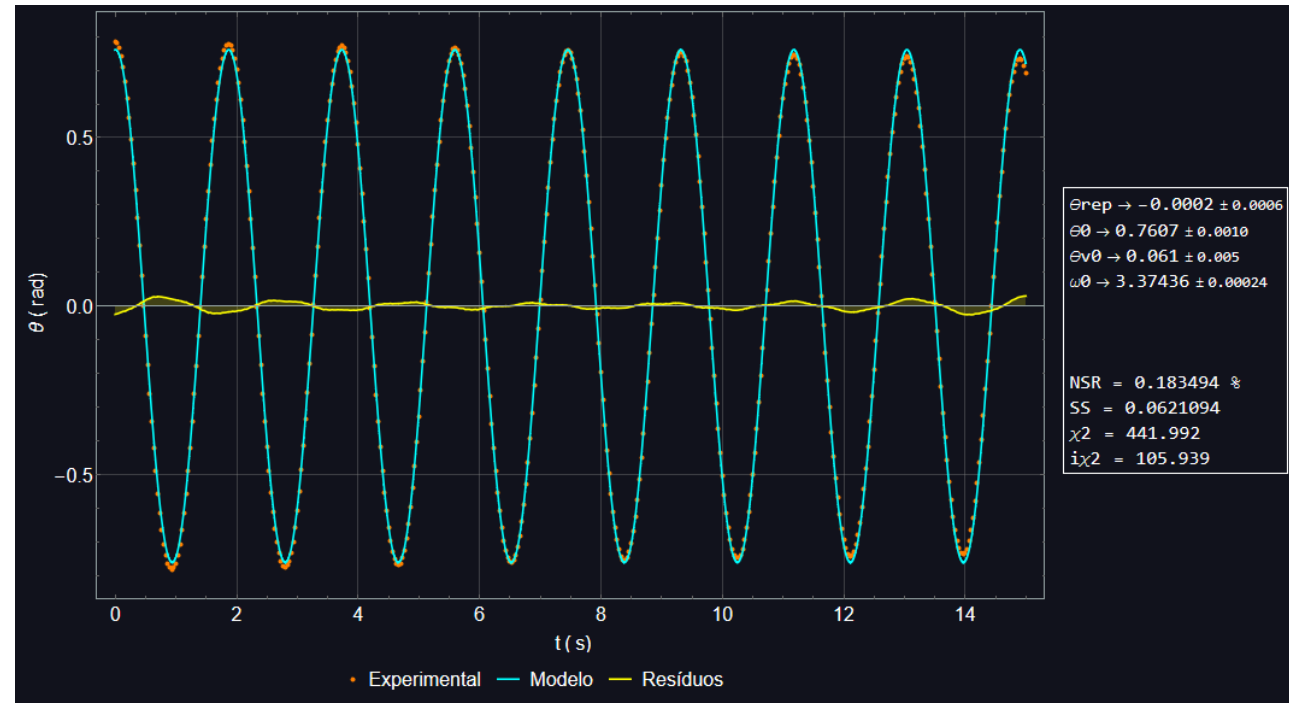
Plot a list of `Around` objects:

```
In[1]:= ListPlot[{Around[1, .1], Around[2, .2], Around[3, .3]}]
```



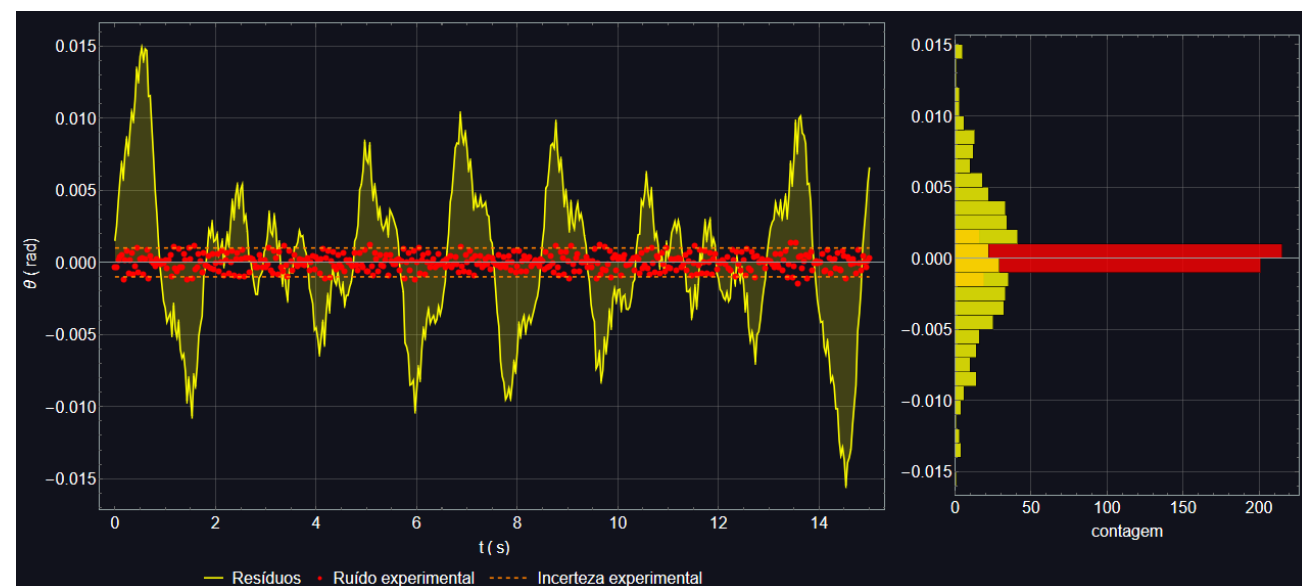
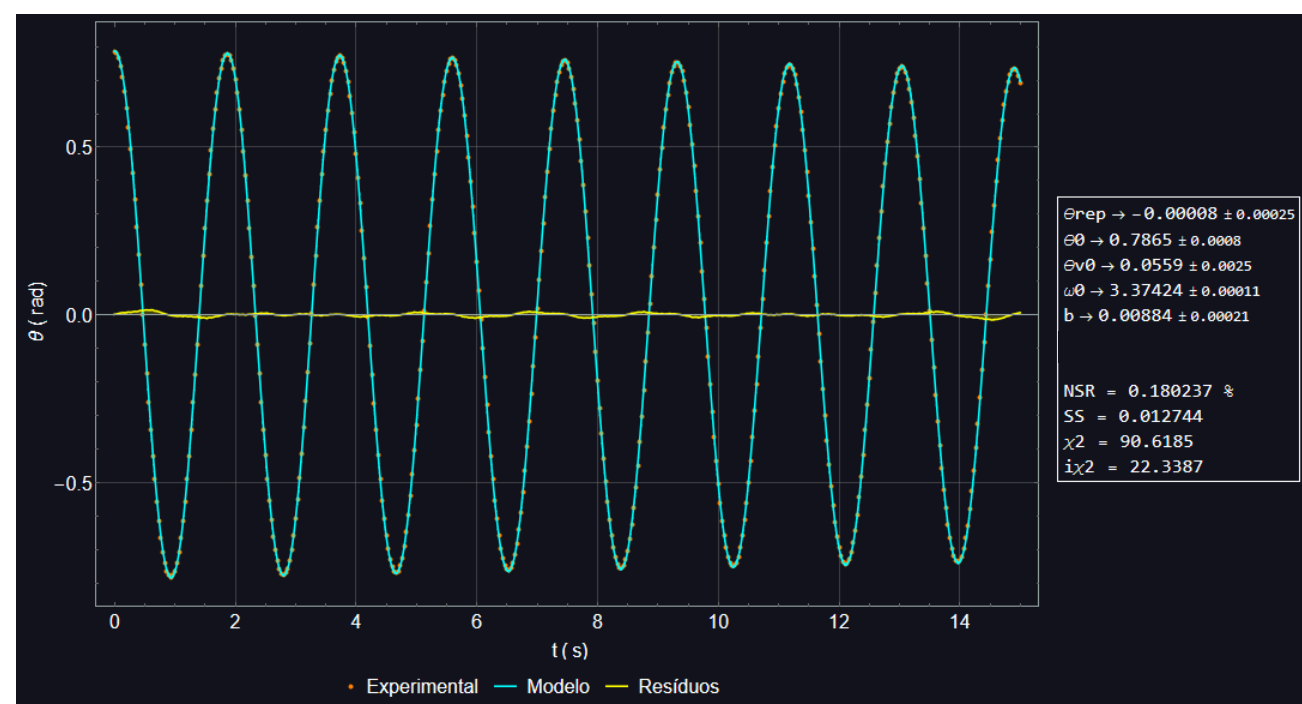
# Dicas

## Análise de resíduos – exemplo I



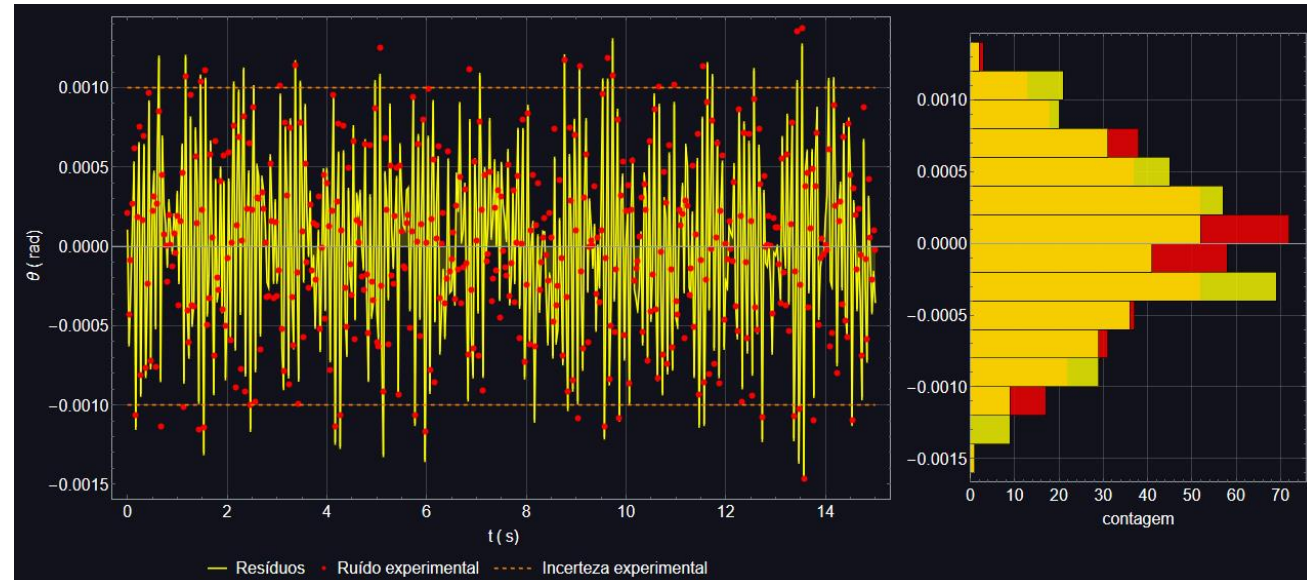
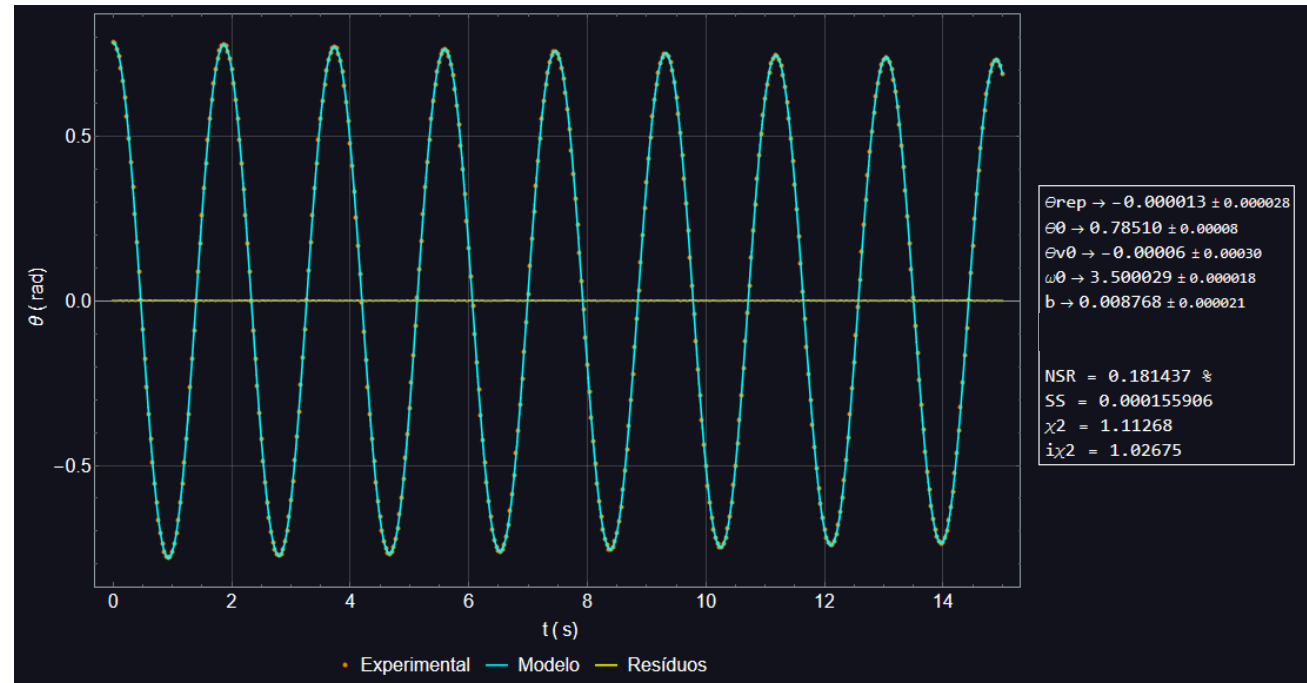
# Dicas

## Análise de resíduos – exemplo II



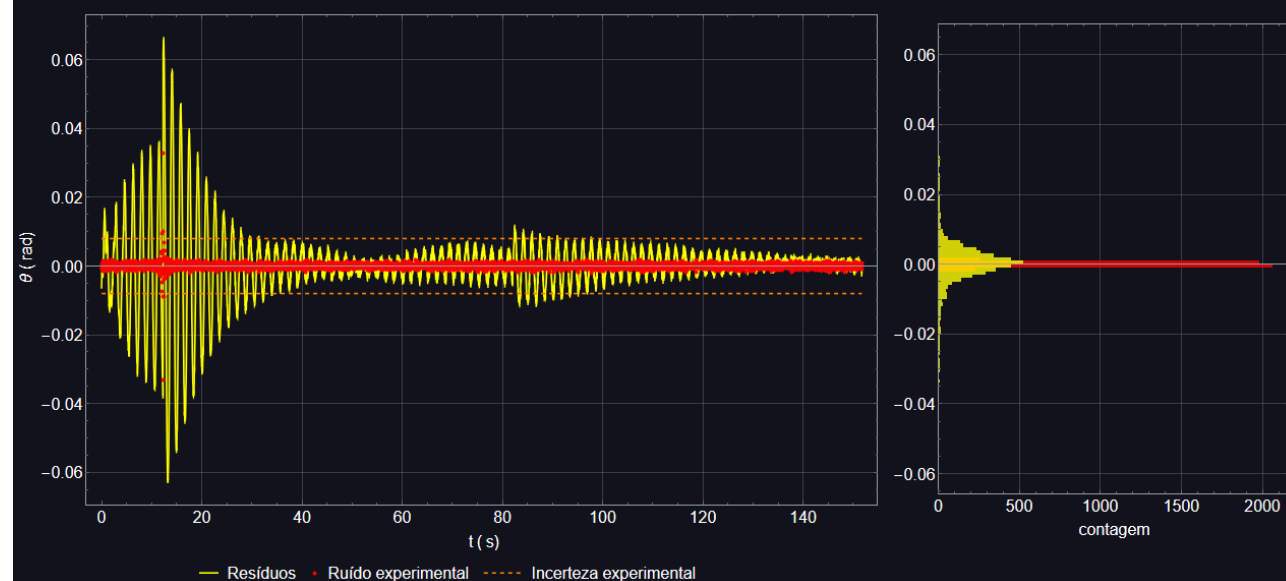
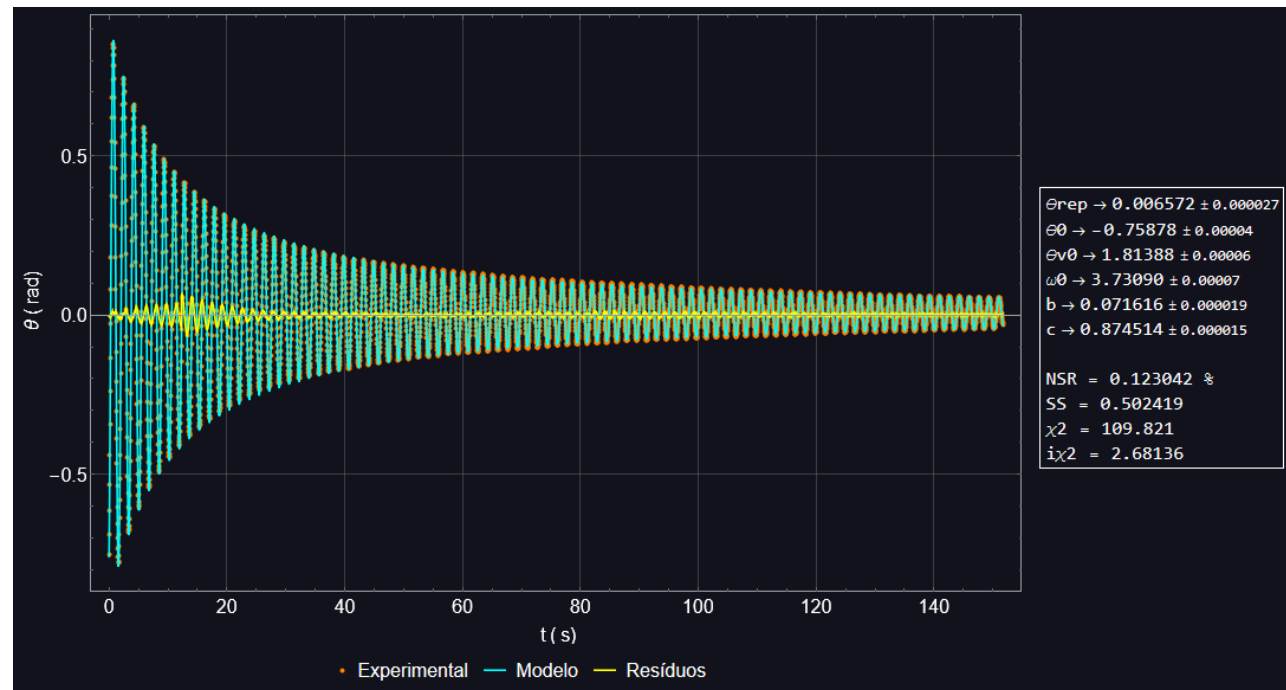
# Dicas

## Análise de resíduos – exemplo III



# Dicas

## Análise de resíduos – exemplo IV



# Dicas

## Análise de resíduos

**Resíduos** = diferença vertical, ponto a ponto, entre a curva experimental e a curva do modelo matemático

**Ruído experimental** = sinal de alta frequência da curva experimental (*noise*), obtido por pelo filtro passa alta

**NSR** = *noise to signal ratio* = razão entre a o RMS do ruído a amplitude máxima dos dados experimentais

**SS** = *sum of squares* = soma dos quadrados dos resíduos

**SN** = soma dos quadrados do ruído experimental

**$\chi^2$**  = *pseudo-Chi-squared* com referência ao ruído =  $SS/SN$

**$i\chi^2$**  = *pseudo-Chi-squared* incluindo incerteza experimental informada =  $(SS/N + inc^2)/(SN/N + inc^2)$

**N** = número de pontos experimentais

**inc** = incerteza informada (desvio padrão médio vertical dos dados)







INSTITUTO  
TECNOLÓGICO  
DE AERONÁUTICA

Introdução à Física  
Experimental

Prof. Douglas Leite  
[www.lpp.ita.br/leite](http://www.lpp.ita.br/leite)

# Ajuste de modelos matemáticos não lineares em dados experimentais usando o Mathematica®

## Aula 2

Vídeo 1 – Introdução e atividades da aula

Vídeo 2 – Tutorial: Básico do Mathematica II

Vídeo 3 – Tutorial: Ajuste de curva no Mathematica



INSTITUTO  
TECNOLÓGICO  
DE AERONÁUTICA

Introdução à Física  
Experimental

Prof. Douglas Leite  
[www.lpp.ita.br/leite](http://www.lpp.ita.br/leite)

# Ajuste de modelos matemáticos não lineares em dados experimentais usando o Mathematica®

## Aula 2

Vídeo 1 – Introdução e atividades da aula

Vídeo 2 – Tutorial: Básico do Mathematica II

Vídeo 3 – Tutorial: Ajuste de curva no Mathematica