Universidade Federal do Rio Grande do Norte (UFRN).
Engenharia da Computação.
Computação Numérica: Projeto Avaliativo 02.
Discente: Marcos Aurélio Tavares Filho
Matrícula: 20240024560

1 Introdução e Contextualização

O projeto 02 visa comparar o ajuste de um conjunto de dados do método dos mínimos quadrados com o uso de um filtro usando funções complexas. Assim, foi dado o seguinte enunciado:

Seja o sistema dinâmico dado pela seguinte função não linear:

$$y[n] = f(y[n-1], y[n-2], u[n], u[n-1])$$

cujo conjunto entrada / saídas é apresentado na tabela **dados.txt**. Implemente um identificador via **Método dos Mínimos Quadrados** e um filtro utilizando **funções no domínio dos complexos**. Em seguida, compare os resultados obtidos utilizando como métricas o **Erro Médio Quadrático** e o **Coeficiente de Correlação de Pearson** (ver Lab 04). Apresente conclusões.

Diante disso, o presente documento tem como objetivo descrever a solução feita por Marcos Aurélio Tavares Filho para o problema a partir do uso da linguagem de programação Python e conta com gráficos e métricas matemáticas para avaliar o melhor modelo de ajuste.

2 Desenvolvimento

2.1 Método dos Mínimos Quadrados (MMQ)

Ciente do enunciado da questão, sabe-se que:

$$x_n = f(x_{n-1}, x_{n-2}, u_n, u_{n-1})$$
$$y_n = f(y_{n-1}, y_{n-2}, u_n, u_{n-1})$$

Em termos de um ajuste de curva com MMQ, escreve-se x_n e y_n como:

$$x_n = w_{0x} \cdot x_{n-2} + w_{1x} \cdot x_{n-1} + w_{2x} \cdot u_n + w_{3x} \cdot u_{n-1}$$
$$y_n = w_{0y} \cdot y_{n-2} + w_{1y} \cdot y_{n-1} + w_{2y} \cdot u_n + w_{3y} \cdot u_{n-1}$$

Ou, em uma equação matricial:

$$x_n = Mx_{n\times 4} \cdot wx_{4\times 1} \qquad y_n = My_{n\times 4} \cdot wy_{4\times 1}$$

Em que:

$$wx_{4\times 1} = pinv(Mx_{n\times 4}) \cdot x_n \qquad wy_{4\times 1} = pinv(My_{n\times 4}) \cdot y_n$$

Feita essa análise e implementado o código, encontra-se os pesos w de cada saída através das pseudo-inversas de Mx e My, de modo que equivalem:

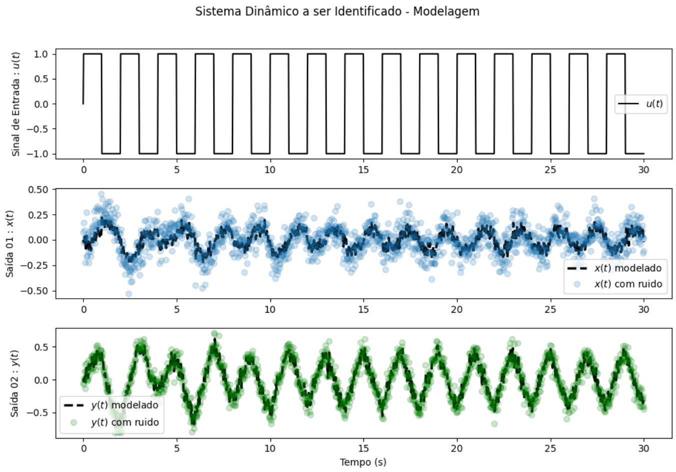
$$wx = [0.243418630.25344686 - 0.03426588 - 0.00187211]$$
$$wy = [0.442742040.47882969 - 0.003740590.03102047]$$

Em posse dos pesos, consegue-se prever o comportamento da curva onde:

$$x_{estimado} = w_{0x} \cdot x_{n-2} + w_{1x} \cdot x_{n-1} + w_{2x} \cdot u_n + w_{3x} \cdot u_{n-1}$$
$$y_{estimado} = w_{0x} \cdot y_{n-2} + w_{1x} \cdot y_{n-1} + w_{2x} \cdot u_n + w_{3x} \cdot u_{n-1}$$

Ao gerar os gráficos resultantes desse processo, tem-se a Figura 1 que descreve como foram os dados captados e como ficou o ajuste via MMQ.

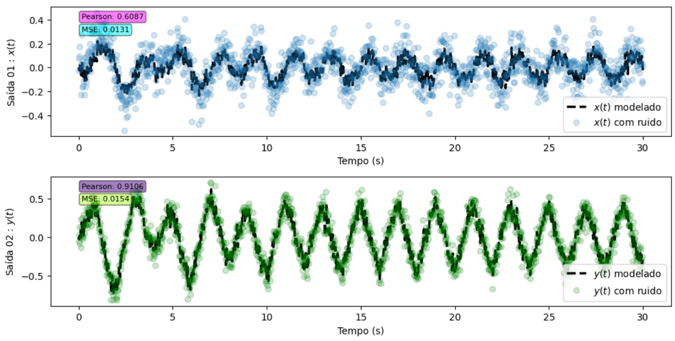
Figura 1: Ajuste via MMQ do conjunto de dados



Fonte: Autoria própria (2025).

Apesar de visualmente ser perceptível que o ajuste é razóavel, usa-se o Erro Médio Quadrado (EMQ) e o Coeficiente de Pearson para poder aferir se a aproximação pode ser classificada como ótima. A Figura 2 mostra os gráficos com os resultados dos cálculos realizados computacionalmente via função da biblioteca *Numpy* em Python.

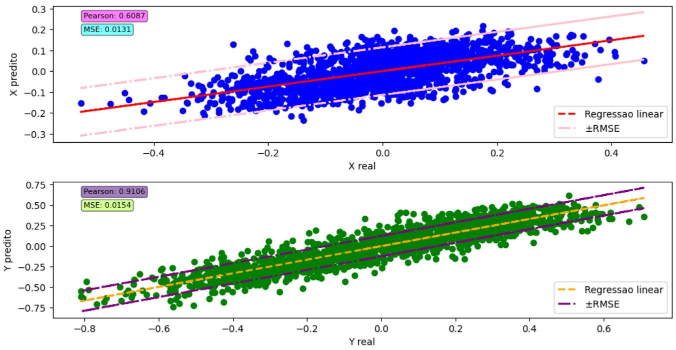
Figura 2: Gráficos do MMQ com métricas



Fonte: Autoria própria (2025).

Como é possível ver acima, os EMQ na saída x e y são $0,131$ e $0,0154$, respectivamente, isso implica que o ajuste classifica-se como bom, uma vez que o erro está entre 0 e 1. No que tange ao coeficiente de Pearson, ambas saídas demonstram forte correlação entre os valores estimados e os valores reais. Em particular, ressalta-se que a saída y foi estimada de forma mais adequada dado o Pearson bem próximo do valor 1. A Figura 3 ilustra a relação entre as saídas e seus valores estimados, além de contar com uma reta de regressão linear elaborada através da função *np.polyfit* do *Numpy*.

Figura 3: Regressão linear do MMQ com métricas



Fonte: Autoria própria (2025).

2.2 Método das Funções Complexas

Para ajustar a curva em termos de funções complexas, necessita-se definir a função de base que será usada. Nesse sentido, tem-se o ajuste via Transformada Discreta de Fourier:

$$X_n = \sum_{k=0}^{N-1} x_k e^{\frac{-j2\pi nk}{Nx}} \qquad Y_n = \sum_{k=0}^{N-1} y_k e^{\frac{-j2\pi nk}{Ny}}$$

No contexto da expressão acima, percebe-se que o X_n e Y_n estão no domínio dos complexos, o que implica que a matriz de transição W é complexa, fato perceptível pelo expoente da exponencial. Ciente disso, evidencia-se que o filtro H será aplicado também em um domínio mais amplo que os reais. Para este projeto, o estabelecimento da janela foi feito de forma experimental com variação até atingir uma representação fidedigna do sinal (30). Além disso, o valor do filtro seguiu metodologia semelhante, em especial, o procedimento foi variar de 0 a 3 com acrésimos de 0.1 até achar o 2 como valor satisfatório.

Com isso, aplica-se o filtro nos dados conforme a equação abaixo.

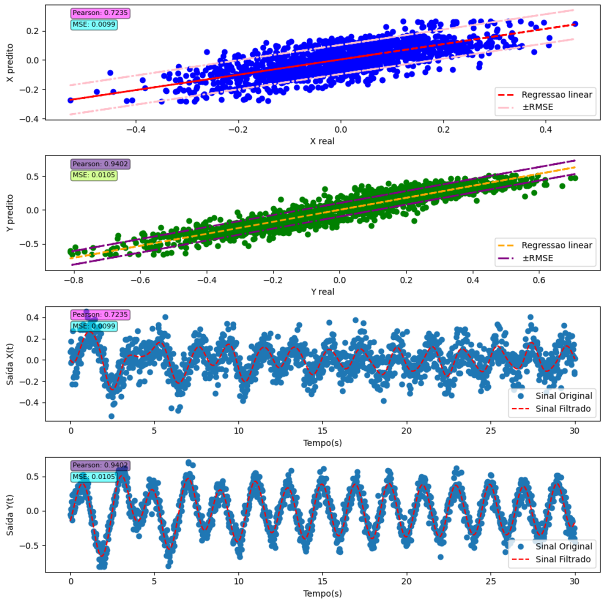
$$X_{filtrado} = Hx \cdot X \qquad Y_{filtrado} = Hy \cdot Y$$

Esse resultado, porém, está nos complexos, para trazer para os reais aplica-se então a matriz hermitiana que corresponde a matriz de transição transposta e conjugada. Assim, nos reais, o sinal filtrado será dado por:

$$x_{filtrado} = Wx^H \cdot X_{filtrado} \qquad y_{filtrado} = Wy^H \cdot Y_{filtrado}$$

Encontrado o sinal filtrado, calculou-se a regressão linear e as métricas de EMQ, coeficiente de Pearson para aferir o desempenho do filtro. A Figura 4 representa os gráficos contendo o valor de Person, os erros e as retas de regressão linear para a saída x e y.

Figura 4: Filtro de conjunto de dados via função complexas



Fonte: Autoria própria (2025).

3 Conclusões

Apresentados os ajustes nas Seções 2.1 e 2.2 e tendo ciência dos coeficientes pelo métodos:

- Mínimos Quadrados
  - Saída X
    - Coeficiente de Pearson: 0,6087
    - Erro Médio Quadrado: 0,0131
  - Saída Y
    - Coeficiente de Pearson: 0,9106
    - Erro Médio Quadrado: 0,0154
- Funções Complexas
  - Saída X
    - Coeficiente de Pearson: 0,7235
    - Erro Médio Quadrado: 0,0099
  - Saída Y
    - Coeficiente de Pearson: 0,9402
    - Erro Médio Quadrado: 0,0105

Percebe-se que **a dispersão dos dados, representado pelo coeficiente de correlação, e o erro apresentam melhores resultados no contexto das funções complexas**. Isso ocorre devido a robustez das funções complexas e a escolha apropriada para a janela e filtro aplicados sobre o sinal no domínio dos complexos.

Ainda, é importante ressaltar que, **apesar da curva ter sido ajustada melhor para as funções complexas nas condições acima, nem sempre isso irá ocorrer**. Inicialmente, neste projeto, foi gasto tempo considerável tentando estabelecer uma janela e filtro para a representação do sinal, com uma performance das complexas inferior ao MMQ, com o tempo, porém, e o aumento dos testes, foi possível estabelecer métricas que preveem o sinal real de forma mais precisa.

Dito isso, conclui-se que, para o estabelecido neste projeto, **o melhor ajuste de curva ocorreu quando foi utilizado as funções complexas.**