

1º Trabalho Computacional

T10097 - Introdução ao Reconhecimento de Padrões

Aluno: Hubert Luz de Miranda

Matrícula: 552798

Fortaleza, 6 de agosto de 2025

Professor Responsável: Prof. Guilherme de Alencar Barreto

Instituição: Universidade Federal do Ceará (UFC)

Objetivos

Os objetivos deste trabalho são:

1. Estimar a matriz de covariância de um conjunto de dados, comparando sistematicamente o tempo de execução de diferentes algoritmos.
2. Avaliar a invertibilidade da matriz de covariância estimada.
3. Inverter e, se necessário, regularizar as matrizes de covariância obtidas.

1 Estimativa da Matriz de Covariância

Antes de prosseguir para a resolução da questão, é válido destacar que os códigos presentes dentro do arquivo **TC01.ipynb** referentes aos métodos de estimativa da matriz de covariância foram adaptações realizadas a partir das funções `mcovar1()`, `mcovar2()`, `mcovar3()` e `mcovar4()` presentes no material de slides do arquivo **classificacao-turmaUFC-2025.pdf**.

1.1 Resultados

Pode-se visualizar os resultados da estimativa da matriz de covariância global no arquivo **TC01.ipynb**, visualmente os métodos apresentam resultados iguais, mas ao realizar a subtração entre as matrizes de covariância estimadas e a matriz nativa, notamos que os métodos 1, 3 e 4 apresentam pequenas diferenças, o que é evidenciado pelas normas das matrizes de diferença calculadas. Embora todas as normas sejam muito próximas de zero (na ordem de 10^{-13} a 10^{-16}), o Método 2 apresenta a menor norma, indicando uma maior similaridade numérica com a função nativa.

1.2 Comentários

Esses resultados confirmam que o Método 2 é o que produz a matriz de covariância numericamente mais próxima à da função nativa, evidenciado pela sua norma da diferença ser ordens de magnitude menor que a dos outros métodos (na casa de 10^{-16}). Isso reforça a hipótese de que a implementação interna da função nativa é baseada em uma abordagem matricial direta e otimizada, similar à do Método 2, que utiliza a matriz de dados centralizada.

2 Análise de Desempenho dos Métodos

2.1 Resultados Numéricos

Tabela 1: Tempo de execução (100 rodadas).

Método	Tempo Médio (s)	Desvio-Padrão (s)
Método 1	2.01e-02	1.66e-03
Método 2	2.92e-04	3.40e-05
Método 3	1.70e-02	5.98e-04
Método 4	2.08e-04	1.20e-05
Função Nativa	3.25e-04	3.90e-05

2.2 Análise Gráfica

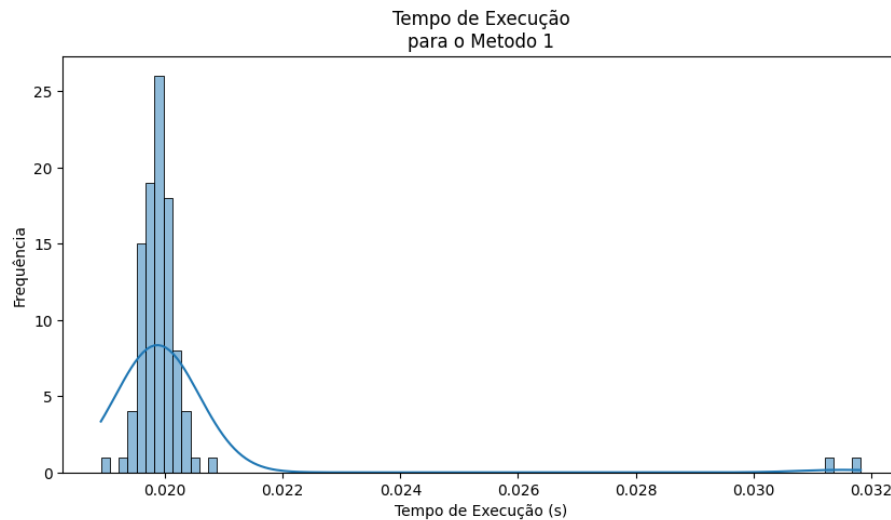


Figura 1: Histograma do tempo de execução do primeiro método

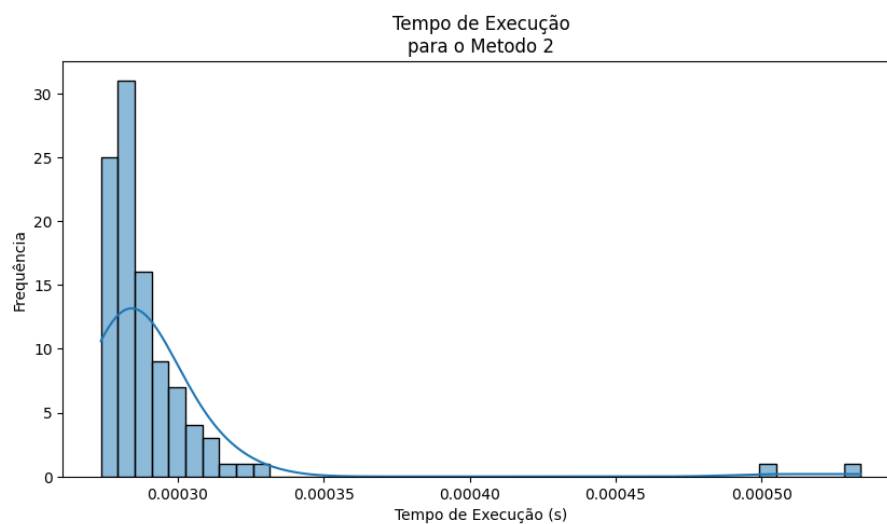


Figura 2: Histograma do tempo de execução do segundo método

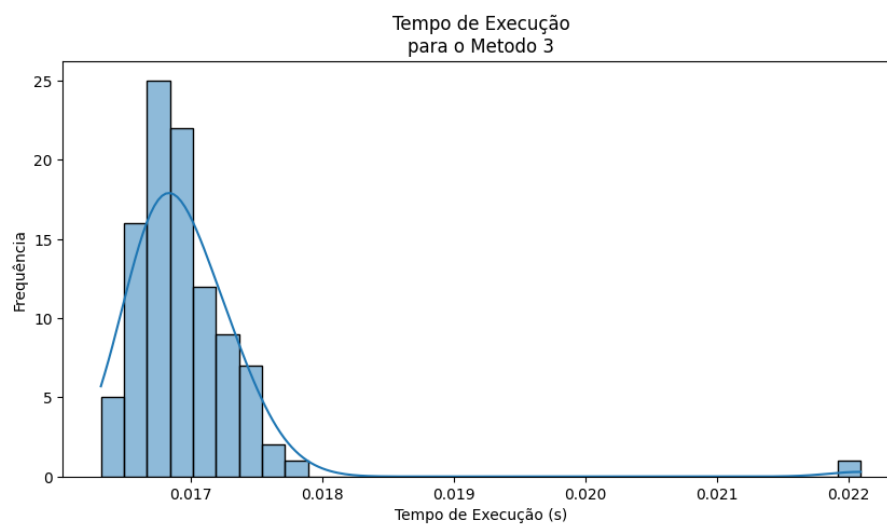


Figura 3: Histograma do tempo de execução do terceiro método

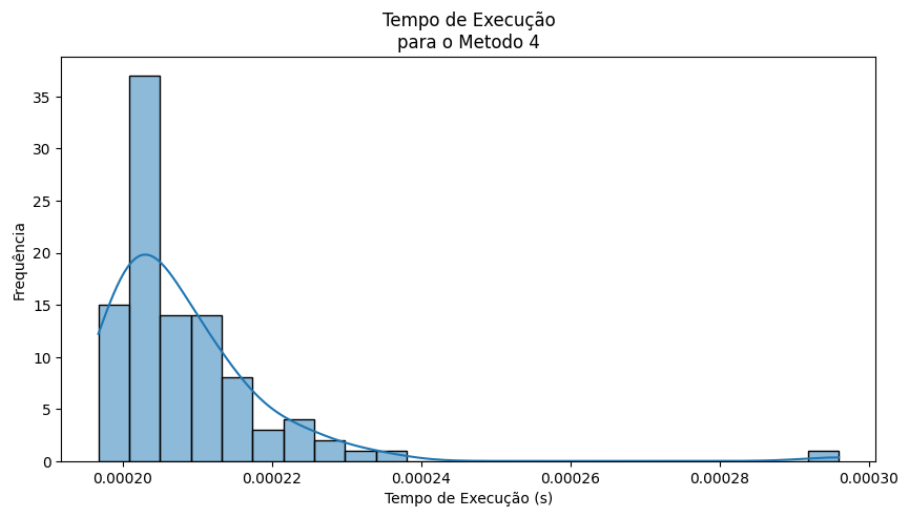


Figura 4: Histograma do tempo de execução do quarto método

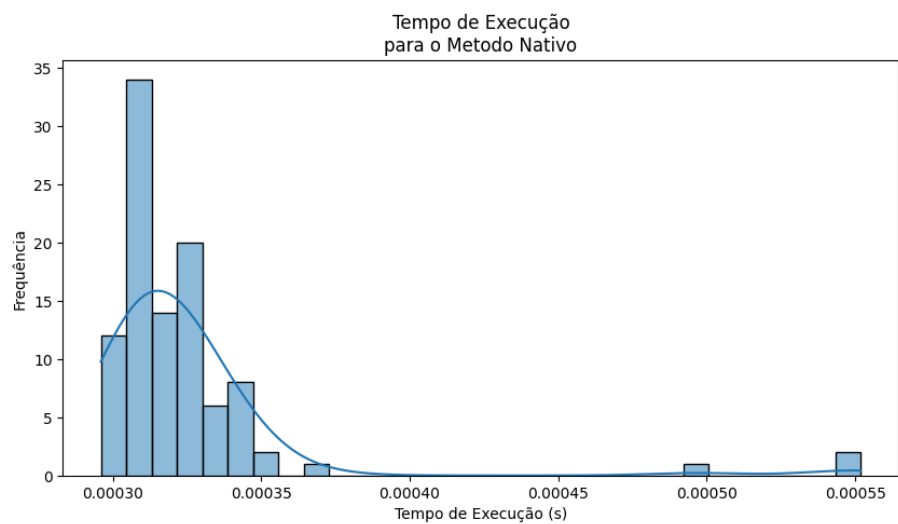


Figura 5: Histograma do tempo de execução do método nativo

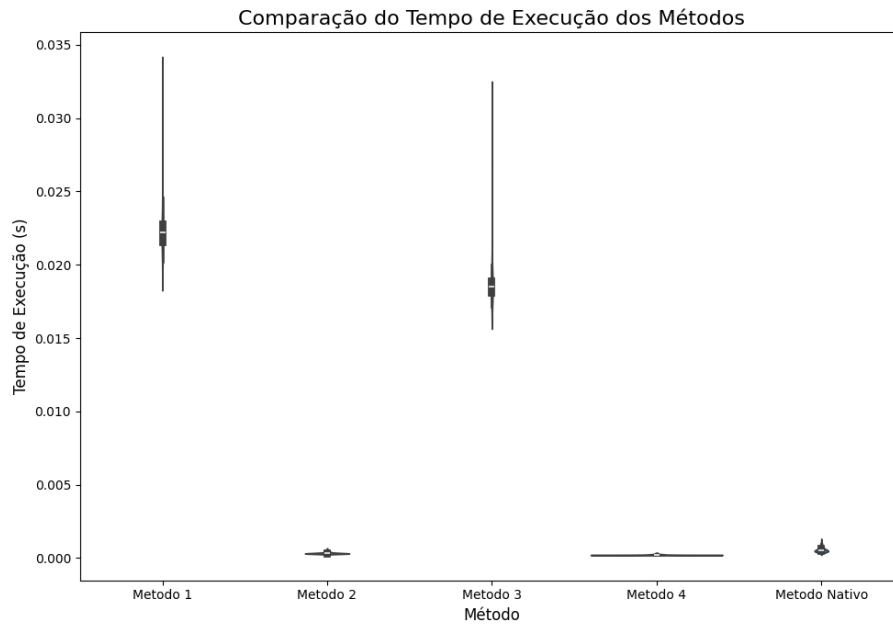


Figura 6: Violin plot para os tempos de execução dos métodos

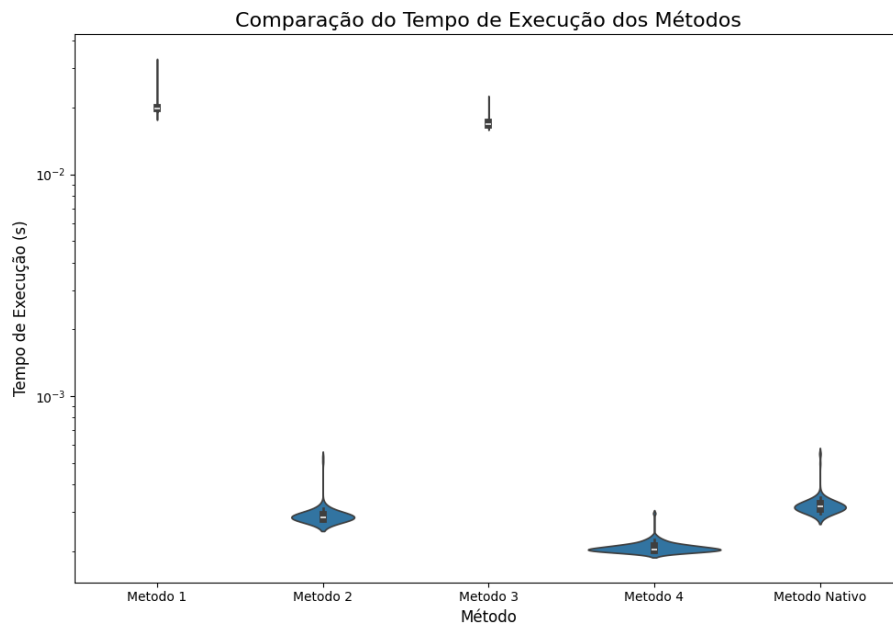


Figura 7: Violin plot para os tempos de execução dos métodos com escala logarítmica

2.3 Comentários

A partir dos resultados obtidos, pode-se observar que os métodos 2, 4 e o nativo possuem tempos de execução significativamente melhores que os métodos 1 e 3. A Figura 6 ilustra a magnitude dessa diferença, enquanto a Figura 7, em escala logarítmica, permite uma análise detalhada da performance dos métodos mais rápidos.

A principal razão para essa discrepância é a implementação. Os métodos 1 e 3, mais lentos, utilizam laços de repetição (`for`) para iterar sobre cada amostra, o que é computa-

cionalmente custoso. Em contraste, os métodos 2 e 4 adotam uma abordagem totalmente matricial (vetorizada), que é altamente otimizada em ambientes computacionais.

Ademais, é visível que o método 4 possui um desvio padrão mais baixo dentre todos, o destacando em relação aos métodos 2 e nativo, o que indica uma maior estabilidade e previsibilidade em seu desempenho. No *violin plot*, a superioridade dos métodos 2, 4 e Nativo é evidente, pois seus gráficos são muito mais compactos verticalmente. A maior dispersão nos resultados dos métodos 1 e 3 é representada por "violinos" mais alongados. É notável que o gráfico do Método 4, apesar de ser o mais rápido, é também o mais largo; isso ocorre porque sua baixíssima variabilidade concentra quase todos os pontos de dados em uma faixa de tempo extremamente estreita, resultando em uma maior densidade (largura) nesse ponto e reforçando sua alta consistência.

3 Análise de Invertibilidade

Para esta questão, o método escolhido para a matriz de covariância global foi o método 4, devido ao seu menor tempo de execução. Posteriormente, as matrizes de covariância por classe foram calculadas e suas invertibilidades foram analisadas.

3.1 Resultados

Tabela 2: Análise de invertibilidade e condicionamento.

Matriz	Posto (rank)	Condicionamento
Global	24	4.87e-02
Move-Forward	24	1.87e-02
Sharp-Right-Turn	24	3.20e-02
Slight-Left-Turn	24	5.77e-03
Slight-Right-Turn	24	5.10e-03

3.2 Comentários

A análise do posto (rank) das matrizes de covariância, apresentada na Tabela, confirma que todas elas são de posto completo (posto = 24). Matematicamente, isso garante que todas as matrizes são invertíveis.

No entanto, o número de condicionamento, também na Tabela, indica que, apesar de invertíveis, as matrizes são mal-condicionadas, com valores relativamente baixos.

A principal consequência de inverter uma matriz mal-condicionada é a instabilidade numérica. Pequenas perturbações nos dados de entrada (como o ruído dos sensores) podem levar a grandes e imprevisíveis alterações na matriz inversa calculada.

4 Inversão e Regularização das Matrizes

Nesta questão, vamos inverter as matrizes e, se necessário, aplicar um método de regularização para evitar qualquer tipo de problema com sua inversão.

4.1 Matrizes Inversas

Ao tentar inverter as matrizes sem aplicar nenhum método de regularização, o sucesso foi obtido, ou seja, todas as inversões foram realizadas.

4.2 Técnica de Regularização

Para melhorar o condicionamento das matrizes, visto que na questão 3 foi evidenciado os seus mal condicionamentos, aplicamos o método de regularização de Tikhonov. Esta técnica consiste em somar uma matriz identidade (I) multiplicada por um pequeno escalar de regularização (λ), que no nosso caso foi igual a 0.01, à matriz de covariância original (C), resultando em uma matriz regularizada $C_{reg} = C + \lambda I$. Com isso, obtivemos os novos valores de condicionamento apresentados na tabela abaixo.

Tabela 3: Comparação do número de condicionamento antes e depois da regularização.

Matriz de Covariância	Condicionamento (Antes)	Condicionamento (Depois)
Global	4.87e-02	4.99e-02
Move-Forward	1.87e-02	2.00e-02
Sharp-Right-Turn	3.20e-02	3.30e-02
Slight-Left-Turn	5.77e-03	8.48e-03
Slight-Right-Turn	5.10e-03	6.25e-03

4.3 Comentários

A tabela acima compara os números de condicionamento antes e depois da aplicação da regularização de Tikhonov. Observa-se uma melhora para todas as matrizes, indicando que a regularização foi bem-sucedida em torná-las numericamente mais estáveis. A melhora foi particularmente notável para as matrizes das classes 'Slight-Left-Turn' e 'Slight-Right-Turn', que eram as mais mal-condicionadas inicialmente, reforçando a eficácia do método para mitigar os problemas causados pelo ruído nos dados.