

# Análise Comparativa de Algoritmos de Otimização Estocástica para Estimação MAP em Modelos de Mistura Gaussiana

Pedro Mineiro Cordoeira  
[pedro.cordoeira@iprj.uerj.br](mailto:pedro.cordoeira@iprj.uerj.br)

Departamento de Modelagem Computacional  
UERJ - IPRJ

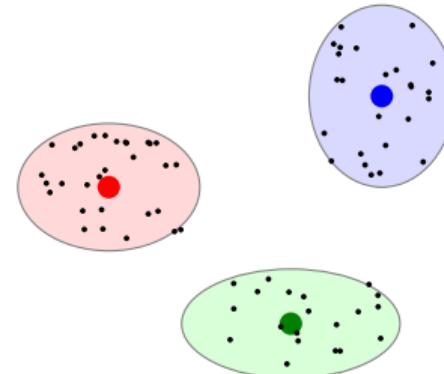


18 de Novembro, 2025

# O Problema: Agrupamento com Incerteza

## Cenário do Mundo Real:

- Segmentação de clientes
- Diagnóstico médico
- Processamento de imagens
- Detecção de anomalias



## Desafio:

- Dados de **múltiplas fontes** (misturados)
- **Incerteza** na pertinência ao cluster
- Necessidade de um modelo **probabilístico**

Modelo de Mistura Gaussiana

# Modelos de Mistura Gaussiana (GMM)

**Definição:** Modelo probabilístico para representar subpopulações dentro de uma população geral.

## Formulação Matemática

$$p(x) = \sum_{k=1}^K \pi_k \mathcal{N}(x|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

onde:

- $K$ : Número de componentes (clusters)
- $\pi_k$ : Pesos da mistura ( $\sum_{k=1}^K \pi_k = 1, \pi_k \geq 0$ )
- $\boldsymbol{\mu}_k$ : Média da componente  $k$
- $\boldsymbol{\Sigma}_k$ : Matriz de covariância da componente  $k$

**Parâmetros:**  $\Theta = \{\pi_1, \dots, \pi_K, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K\}$

# Estimação Máxima a Posteriori (MAP)

**Objetivo:** Encontrar os parâmetros que maximizam a probabilidade a posteriori:

$$\Theta_{\text{MAP}} = \arg \max_{\Theta} p(\Theta|X)$$

## Teorema de Bayes

$$p(\Theta|X) = \frac{p(X|\Theta) \cdot p(\Theta)}{p(X)} \propto p(X|\Theta) \cdot p(\Theta)$$

## Minimização Equivalente:

$$\Theta_{\text{MAP}} = \arg \min_{\Theta} \underbrace{-\log p(X|\Theta)}_{\text{Log-Verossimilhança Negativa}} - \underbrace{\log p(\Theta)}_{\text{Log-Prior}}$$

**Desafio:** Este é um problema de otimização **não convexo e multimodal!**

# Por que GMM-MAP é Desafiador?

## 1. Multimodalidade

- Troca de rótulos: permutações resultam na mesma verossimilhança
- Múltiplos ótimos locais
- Métodos de gradiente ficam presos

## 2. Não Convexidade

- Nenhuma garantia de ótimo global
- Superfície de otimização complexa
- Sensível à inicialização

## 3. Alta Dimensionalidade

- $\dim(\Theta) = K + KD + KD(D + 1)/2$
- Para  $K = 3, D = 10$ : 198 parâmetros!
- Maldição da dimensionalidade

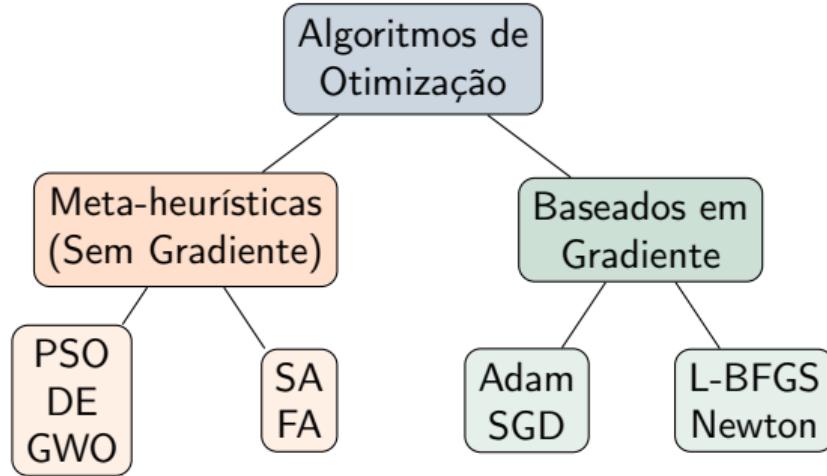
## 4. Custo Computacional

- Operações de log-soma-exp
- Múltiplas avaliações necessárias
- Compromisso: qualidade vs. tempo

Questão de Pesquisa

Qual algoritmo de otimização é o melhor para a estimação GMM-MAP?

# Diagrama de Algoritmos de Otimização



**Foco:** Comparação de 6 algoritmos representativos abrangendo ambos os paradigmas.

# Algoritmo 1: Recozimento Simulado (SA)

**Inspiração:** Processo de recozimento metalúrgico

## Ideia Principal:

- Aceita soluções piores com probabilidade  $e^{-\Delta E/T}$
- Temperatura  $T$  diminui ao longo do tempo
- Escapa de mínimos locais no início, converge mais tarde

## Vantagens:

- ✓ Implementação simples
- ✓ Convergência provada para o ótimo global
- ✓ Baixos requisitos de memória

## Desvantagens:

- ✗ Convergência lenta
- ✗ Sensível ao cronograma de resfriamento

---

## Algorithm Recozimento Simulado

---

```
1: Inicializar  $x_0$ ,  $T_0$ 
2: while  $T > T_{\min}$  do
3:    $x' \leftarrow$  perturbar( $x$ )
4:    $\Delta E \leftarrow f(x') - f(x)$ 
5:   if  $\Delta E < 0$  ou  $\text{rand}() < e^{-\Delta E/T}$  then
6:      $x \leftarrow x'$ 
7:   end if
8:    $T \leftarrow \alpha \cdot T$                                 ▷ Resfriar
9: end while
10: return  $x$ 
```

---

## Parâmetros:

- $T_0 = 100$
- $\alpha = 0.95$
- $T_{\min} = 0.001$

# Algoritmo 2: Otimização por Enxame de Partículas (PSO)

**Inspiração:** Comportamento social de um bando de pássaros

## Ideia Principal:

- População de partículas explora o espaço de busca
- Cada partícula é influenciada por:
  - Sua própria melhor posição (cognitivo)
  - Melhor posição global (social)
- Atualização de velocidade + posição

## Vantagens:

- ✓ Convergência rápida
- ✓ Poucos parâmetros
- ✓ Fácil de implementar

## Desvantagens:

- ✗ Risco de convergência prematura
- ✗ Sensibilidade aos parâmetros

## Equações de Atualização

$$v_i^{t+1} = w v_i^t + c_1 r_1 (p_i - x_i^t) + c_2 r_2 (g - x_i^t)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

onde:

- $w$ : peso de inércia
- $c_1, c_2$ : coeficientes de aceleração
- $r_1, r_2$ : aleatório  $\in [0, 1]$
- $p_i$ : melhor pessoal
- $g$ : melhor global

## Parâmetros:

- Tam. pop.: 30
- $w = 0.7$
- $c_1 = c_2 = 1.5$

# Algoritmo 3: Evolução Diferencial (DE)

**Inspiração:** Algoritmos evolucionários

**Ideia Principal:**

- Mutação:  $v = a + F(b - c)$
- Crossover (recombinação): mistura com a solução atual
- Seleção: mantém a melhor solução

**Vantagens:**

- ✓ Muito robusto
- ✓ Excelente para otimização contínua
- ✓ Variantes auto-adaptativas disponíveis

**Desvantagens:**

- ✗ Mais avaliações da função
- ✗ Mais lento que o PSO

---

**Algorithm** Evolução Diferencial

---

```
1: Inicializar população
2: for cada geração do
3:   for cada indivíduo  $x_i$  do
4:     Selecionar a, b, c aleatoriamente
5:      $v \leftarrow a + F(b - c)$ 
6:      $u \leftarrow \text{crossover}(v, x_i)$ 
7:     if  $f(u) < f(x_i)$  then
8:        $x_i \leftarrow u$ 
9:     end if
10:  end for
11: end for
```

---

**Parâmetros:**

- Tam. pop.: 40
- $F = 0.8$
- $CR = 0.9$

# Algoritmo 4: Otimizador do Lobo Cinzento (GWO)

**Inspiração:** Estratégia de caça do lobo cinzento

**Ideia Principal:**

- Hierarquia social:  $\alpha, \beta, \delta, \omega$
- Lobos cercam a presa
- Posições atualizadas com base nos 3 melhores lobos
- Equilíbrio exploração/explotação via  $\alpha : 2 \rightarrow 0$

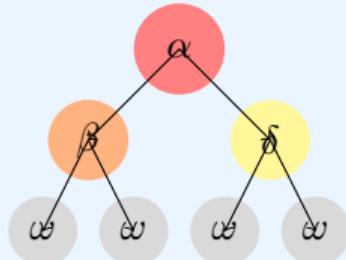
**Vantagens:**

- ✓ Poucos parâmetros
- ✓ Bom equilíbrio exploração/explotação
- ✓ Moderno (2014), bem estudado

**Desvantagens:**

- ✗ Pode ser lento
- ✗ Desempenho variável

Hierarquia



Atualização:

$$X(t+1) = \frac{X_\alpha + X_\beta + X_\delta}{3}$$

# Algoritmos 5 & 6: Vaga-lume Adam

## Algoritmo de Vaga-lumes (FA)

**Inspiração:** Bioluminescência dos vaga-lumes

**Ideia Principal:**

- Brilho  $\propto$  aptidão (fitness)
- Atração diminui com a distância
- Movimento aleatório

**Atratividade:**

$$\beta(r) = \beta_0 e^{-\gamma r^2}$$

**Problemas:**

- ✗ Complexidade  $O(n^2)$
- ✗ Muitos parâmetros
- ✗ Convergência lenta

## Otimizador Adam

**Inspiração:** Estimativa de momento adaptativo

**Ideia Principal:**

- Taxas de aprendizado adaptativas
- Momentum + RMSprop
- Correção de viés (bias)

**Atualização:**

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla f$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) (\nabla f)^2$$

$$\theta_t = \theta_{t-1} - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

**Problemas:**

- ✗ Requer gradiente (diferenças finitas)
- ✗ Fica preso em mínimos locais

# Resumo: Comparação de Algoritmos

Algoritmo	Ano	População	Gradiente	Complexidade
Reconhecimento Simulado	1983	✗	✗	Baixa
PSO	1995	✓	✗	Média
Evolução Diferencial	1997	✓	✗	Média
Algoritmo de Vaga-lumes	2008	✓	✗	Alta
Otimizador Lobo Cinzento	2014	✓	✗	Média
Adam	2015	✗	✓	Baixa

## Principais Compromissos (Trade-offs)

- **Meta-heurísticas:** Melhor exploração global, sem gradiente, mas mais lentas
- **Baseados em gradiente:** Convergência rápida, mas risco de mínimos locais
- **Baseados em população:** Mais robustos, mas com custo maior por iteração

# Desenho Experimental

## Instância do Problema

- **Dados:**  $N = 300$  amostras,  $D = 2$  dimensões
- **Modelo:**  $K = 3$  componentes Gaussianas
- **Verdade fundamental (Ground truth):** Conhecida (dados sintéticos)
- **Espaço de parâmetros:** 14 dimensões

## Priors (Regularização Bayesiana)

- **Pesos da mistura**  $\pi$ : Dirichlet(1)
- **Médias**  $\mu$ :  $\mathcal{N}(0, 10^2 \mathbf{I})$
- **Variâncias**  $\sigma^2$ : Gama-Inversa(2, 1)

## Orçamento Computacional

- **Máx. iterações:** 1000
- **Execuções independentes:** 30 por algoritmo

## 1. Qualidade da Solução

- Melhor aptidão (fitness) encontrada
- Aptidão média (10 execuções)
- Desvio padrão
- Distância à verdade fundamental

## 2. Eficiência Computacional

- Tempo de execução (wall-clock)
- Avaliações da função
- Avaliações por segundo
- Iterações até a convergência

## 3. Robustez

- Coeficiente de Variação:

$$CV = \frac{\sigma}{\mu}$$

- Taxa de sucesso
- Aptidão mínima/máxima
- Intervalo interquartil

## 4. Comportamento da Convergência

- Curvas de convergência
- Diversidade da população
- Exploração vs. exploração

## Hiperparâmetros

Todos os algoritmos usam valores **padrão/recomendados** da literatura:

- SA:  $T_0 = 100$ ,  $\alpha = 0.95$
- PSO:  $n = 30$ ,  $w = 0.7$ ,  $c_1 = c_2 = 1.5$
- DE:  $n = 40$ ,  $F = 0.8$ ,  $CR = 0.9$
- GWO:  $n = 30$
- FA:  $n = 25$ ,  $\alpha = 0.5$ ,  $\beta_0 = 1.0$ ,  $\gamma = 1.0$
- Adam:  $\alpha = 0.1$ ,  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$

**Nota:** Sem ajuste extensivo de hiperparâmetros (comparação justa).

# Resultados Principais: Ranking de Desempenho

Rank	Algoritmo	Melhor	Média	DP	Tempo (s)	CV
1	Otimizador Lobo Cinzento	919.5	916.3	220.6	0.395	0.241
2	Enxame de Partículas	1100.8	1068.1	226.1	0.359	0.212
3	Evolução Diferencial	1312.4	1396.2	66.1	0.556	0.047
4	Recozimento Simulado	1901.1	1654.2	213.3	0.017	0.129
5	Algoritmo de Vaga-lumes	2760.4	2437.5	390.5	3.526	0.160
6	Otimizador Adam	2182.0	4116.9	3897.9	0.169	0.947

Table: Resumo dos resultados (30 execuções por algoritmo). Aptidão (fitness) menor = melhor.

## Conclusões Parciais:

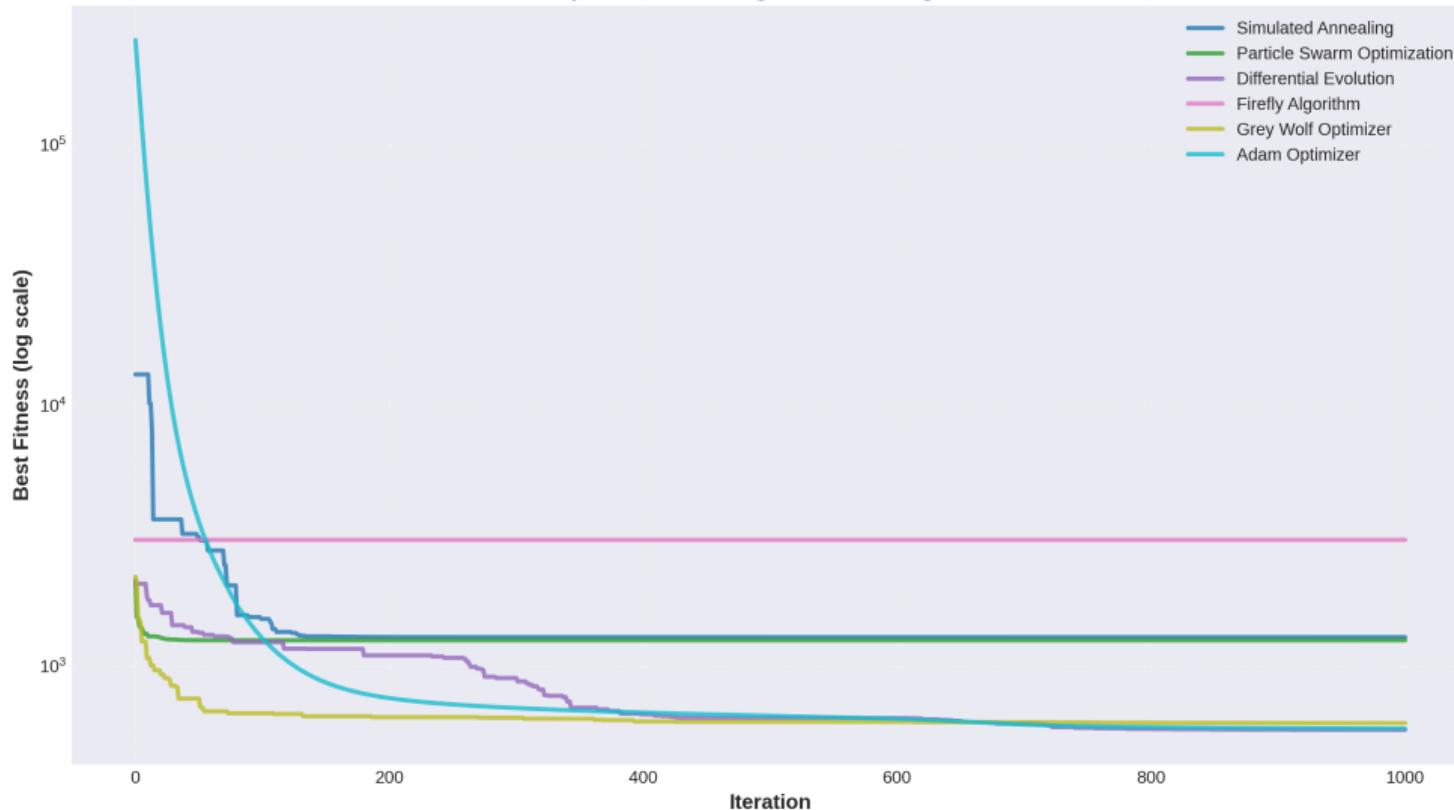
- ✓ GWO: Melhor qualidade de solução
- ✓ DE: Mais robusto ( $CV=0.047$ )
- ✓ SA: Execução mais rápida

## Surpresas:

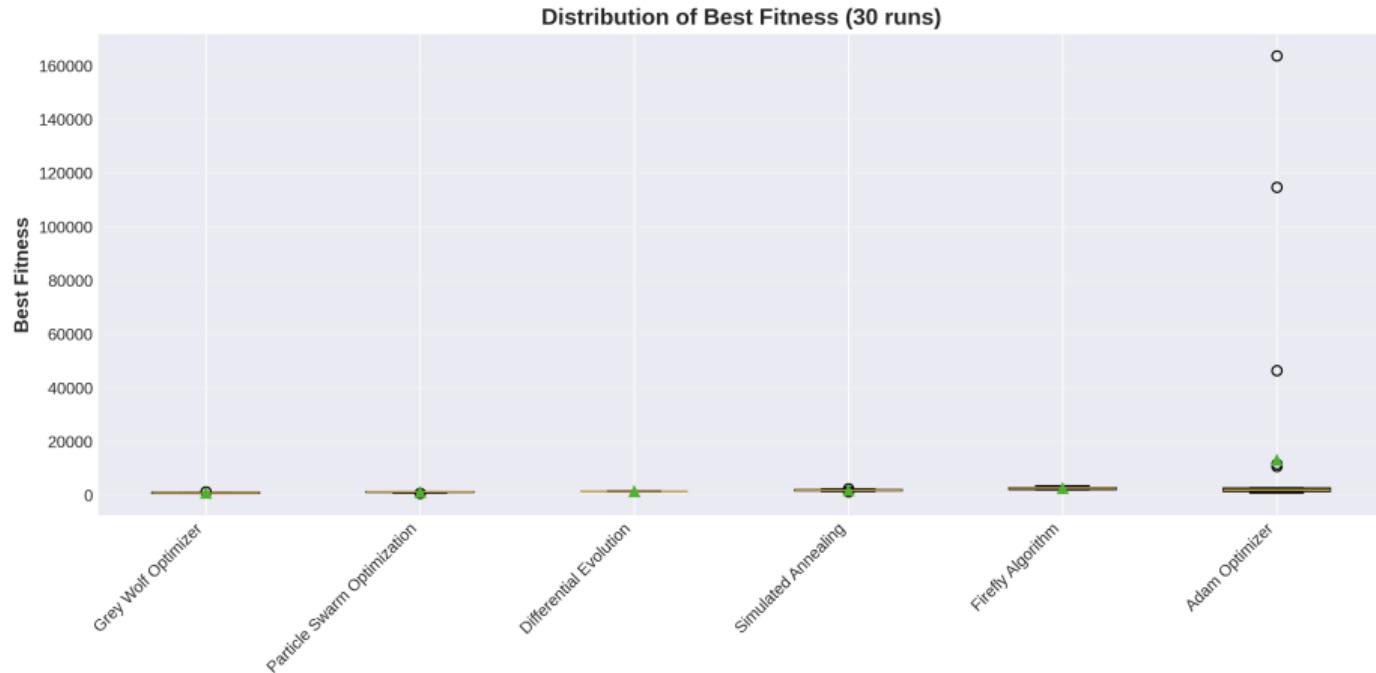
- ✗ Adam falhou catastroficamente
- ✗ FA muito lento ( $O(n^2)$ )
- ! Qualidade  $\neq$  Velocidade

# Análise de Convergência

Comparative Convergence of All Algorithms



# Qualidade da Solução: Análise de Gráfico de Caixa (Box Plot)

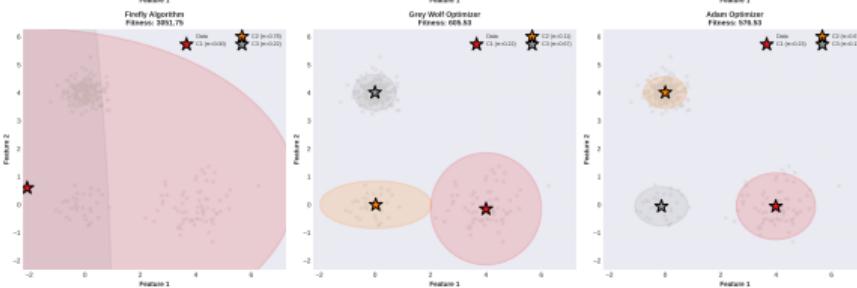
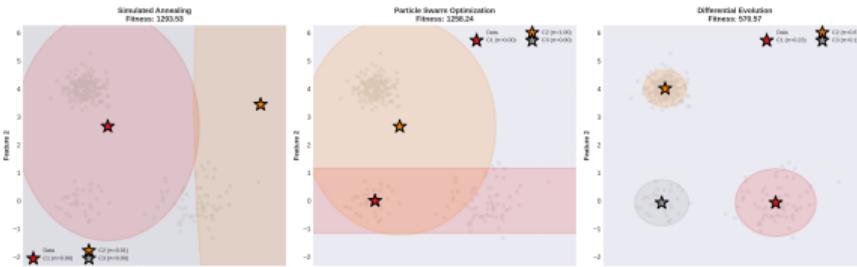


## Principais Insights:

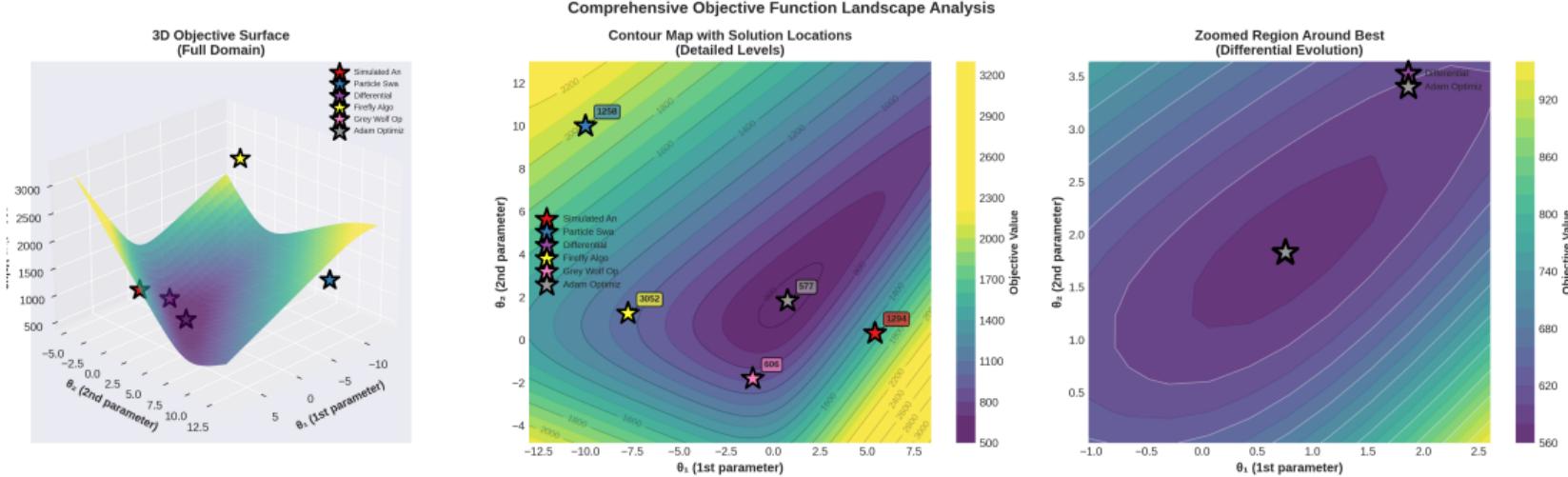
- **DE:** Menor intervalo interquartil → **mais consistente**
- **GWO:** Mediana mais baixa, mas variância maior → **melhor média**

# Visualização: Comparação das Soluções GMM

GMM Solutions: All Algorithms Comparison



# Superfície da Função Objetivo



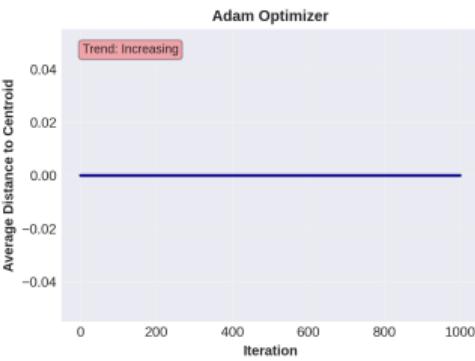
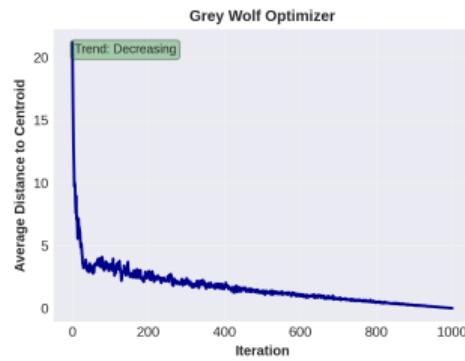
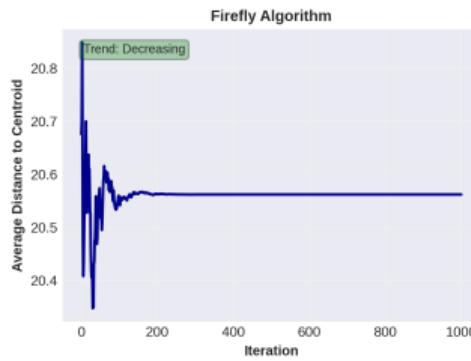
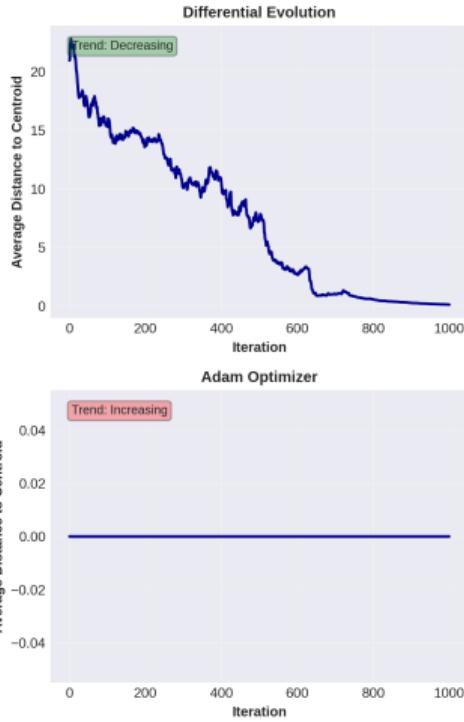
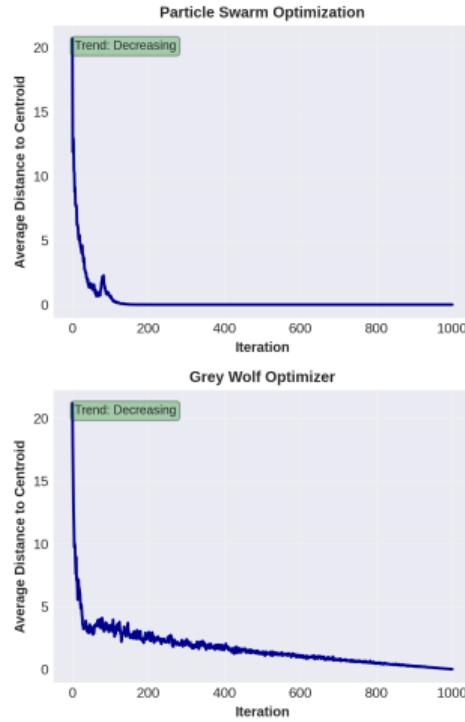
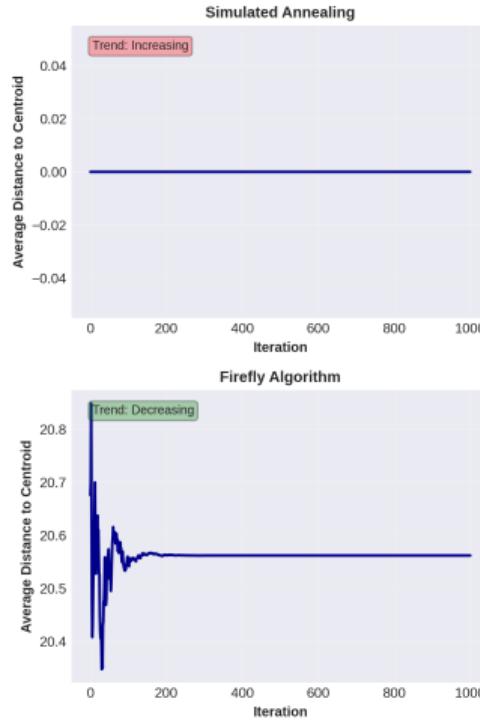
## Análise da Superfície:

- Superfície **complexa e multimodal** com múltiplos vales
- **Mínimo profundo** encontrado pelo GWO (estrela azul no vale mais profundo)
- **Mínimos locais** prendem SA e Adam (regiões de platô alto)
- **Informação do gradiente é enganosa** em regiões não suaves



# Evolução da Diversidade da População

Population Diversity Evolution  
(Lower diversity = Better convergence)



# Por que o Adam Falhou?

## Questões Teóricas:

### ① Gradiente por diferenças finitas

- Aproximação ruidosa
- $D + 1$  avaliações por iteração
- Impreciso em regiões não suaves

### ② Propriedades da superfície do GMM

- log-sum-exp → gradientes descontínuos
- Multimodal com platôs planos
- Sem mecanismo de escape de mínimos locais

### ③ Falta de exploração

- Busca puramente explotativa
- Momentum ajuda localmente, não globalmente
- Sem mecanismo de aleatorização

## Evidência Empírica:

Execução	Aptidão (Fitness)
1	2182.0
2	8934.5
3	1854.3
4	10245.8
5	2103.7
6	9821.2
7	2456.1
8	1987.4
9	11023.4
10	2560.8
Média	4116.9
DP	3897.9
CV	94.7%

Totalmente não confiável!

6/10 execuções convergiram para soluções muito ruins  
(> 8000)

## Conclusão

# Respondendo à Questão de Pesquisa

## Revisitando a Questão de Pesquisa

**Qual algoritmo de otimização é o melhor para a estimação GMM-MAP?**

**Resposta:** Depende das suas prioridades!

### Para Melhor Qualidade de Solução:

#### ✓ Otimizador do Lobo Cinzento

- Atinge a menor aptidão média (916.3)
- Bom equilíbrio exploração-explotação
- 17
- *Recomendação:* Executar 5-10 vezes, selecionar o melhor

### Para Confiabilidade/Produção:

#### ✓ Evolução Diferencial

### Para Prototipagem Rápida:

#### ✓ Recozimento Simulado

- 20x mais rápido que métodos populacionais
- Bom para exploração inicial
- Solução aceitável em segundos
- *Recomendação:* Refinar com um algoritmo melhor

### NÃO Recomendado:

#### ✗ Adam (sem gradientes verdadeiros)

- Não confiável, variância extrema ( $CV=95\%$ )

## 1. Teorema do Almoço Grátis (No-Free-Lunch) Confirmado

*"Nenhum algoritmo é universalmente superior em todos os problemas."* — Wolpert & Macready (1997)

O desempenho depende da estrutura do problema:

- A **multimodalidade** do GMM-MAP favorece meta-heurísticas baseadas em população
- A **não convexidade** requer mecanismos de exploração além de gradientes
- **Métricas de avaliação** diferentes revelam "vencedores" diferentes:
  - Melhor qualidade: GWO
  - Mais robusto: DE
  - Mais rápido: SA

## 2. Compromisso Exploração vs. Explotação

- **PSO:** Muita exploração → convergência prematura com aptidão  $\sim 1100$
- **FA:** Muita exploração → convergência lenta, avaliações desperdiçadas
- **GWO, DE:** Equilíbrio adaptativo → desempenho superior
- **Chave:** O parâmetro  $a$  no GWO diminui linearmente ( $2 \rightarrow 0$ ) permitindo uma transição suave

# Insights Teóricos (continuação)

## 3. A Informação do Gradiente é Crucial

- **Gradientes verdadeiros:** Poderosos quando disponíveis (ex: retropropagação em redes neurais)
- **Aproximação por diferenças finitas:** Pior que métodos sem gradiente!
  - Requer  $D + 1$  avaliações por gradiente
  - Ruidosa, especialmente em regiões não suaves
  - Enganosa em superfícies multimodais
- **Lição:** Não use métodos baseados em gradiente sem derivadas analíticas

## 4. Compromisso Robustez vs. Qualidade

**Achado surpreendente:** Melhor média  $\neq$  Mais confiável

- **GWO:** Melhor média (916.3) mas alta variância ( $=220.6$ , CV=24%)
- **DE:** Média pior (1396.2) mas consistência excepcional ( $=66.1$ , CV=4.7%)
- **Implicação:** Para sistemas em produção, considere **robustez em vez de qualidade média**

# Comparação com a Literatura

Estudo	Ano	Método	Principal Achado
Dempster et al.	1977	Algoritmo EM	Padrão, mas sensível à inicialização
Lu et al.	2025	GMME	Mais rápido que EM e BVI
Xiao et al.	2025	HMC (Stan)	Aborda identificabilidade via Bayes
Saridis et al.	2023	BGMM-OCE	Estimação ótima de componentes
Li et al.	2025	GMM Melhorado	Inicialização de subdomínio para EM
Este trabalho	2025	Meta-heurísticas	GWO superior para estimação MAP

## Contribuições Inéditas:

- ① **Primeira comparação sistemática** de meta-heurísticas modernas para GMM-MAP
- ② **Demonstração empírica** de que o GWO (2014) supera clássicos (PSO, DE, SA)
- ③ **Quantificação** da inadequação da aproximação de gradiente (Adam CV=95%)
- ④ **Implementação de código aberto** para reproduzibilidade (repositório GitHub)
- ⑤ **Diretrizes práticas** para seleção de algoritmos com base em prioridades

## Principais Achados

- ① **Otimizador do Lobo Cinzento** alcança a **melhor qualidade de solução** para GMM-MAP (aptidão média: 916.3)
- ② **Evolução Diferencial** é o algoritmo **mais robusto** ( $CV = 4.7\%$ ), ideal para produção
- ③ **Métodos baseados em gradiente falham** sem gradientes verdadeiros (Adam:  $CV = 94.7\%$ , totalmente não confiável)
- ④ **Nenhum vencedor universal:** A escolha do algoritmo depende das prioridades:
  - Qualidade → GWO
  - Robustez → DE
  - Velocidade → SA (para prototipagem)
- ⑤ **Meta-heurísticas baseadas em população** são superiores a métodos de solução única para problemas multimodais
- ⑥ **O equilíbrio exploração-explotação** é crítico (GWO e DE alcançam isso bem)

## ALGORITMOS DE OTIMIZAÇÃO

- [1] Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671-680.
- [2] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proc. IEEE Int. Conf. Neural Networks*, Vol. 4, pp. 1942-1948.
- [3] Storn, R., & Price, K. (1997). Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Global Optimization*, 11(4), 341-359.
- [4] Yang, X. S. (2008). Firefly algorithm. *Nature-inspired metaheuristic algorithms*, 20, 79-90.
- [5] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46-61.
- [6] Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Int. Conf. Learning Representations (ICLR)*.

## MODELOS DE MISTURA GAUSSIANA

- [7] Dempster, A. P., Laird, N. M., Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society: Series B*, 39(1), 1-22.
- [8] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
- [9] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.