

MÉTODOS E MODELOS AVANÇADOS EM CIÊNCIA DE DADOS

Aula 07 - Inteligência de Enxames
(*Swarm Intelligence*)

Prof. Rafael G. **Mantovani**

Roteiro



- 1** Introdução
- 2** *Swarm Intelligence*
- 3** *PSO*
- 4** Síntese / Próximas Aulas
- 5** Referências

Roteiro

- 1** Introdução
- 2** *Swarm Intelligence*
- 3** *PSO*
- 4** Síntese / Próximas Aulas
- 5** Referências

Introdução

□ Otimização

- importante área de estudo
- virtualmente aplicável em diferentes áreas de estudo
- assim também é na Ciência de Dados/*ML*

Introdução



Exemplos?

Introdução

□ Gerais:

- encontrar melhor rota de robôs
- planejamento de tarefas
- corte e empacotamento

□ ML / DS

- ajuste de hiperparâmetros (HPs)
- seleção de atributos (*feature selection*)

Introdução

□ Problema de otimização

- minimização / maximização
- facilmente convertidos entre si

$$\min_x f(x) \leftrightarrow \max_x [-f(x)]$$
$$\max_x f(x) \leftrightarrow \min_x [-f(x)]$$

Introdução

□ Problema de otimização

- minimização / maximização
- facilmente convertidos entre si

$$\min_x f(x) \leftrightarrow \max_x [-f(x)]$$
$$\max_x f(x) \leftrightarrow \min_x [-f(x)]$$

$f(x)$ é função objetivo

Introdução

- **Diferentes técnicas/algoritmos**
 - Simples
 - Meta-heurísticas (métodos estocásticos)

Adaptação

Introdução

- **Diferentes técnicas/algoritmos**

- Simples
- Meta-heurísticas (métodos estocásticos)

Adaptação

Aleatoriedade

Introdução

- **Diferentes técnicas/algoritmos**

- Simples
- Meta-heurísticas (métodos estocásticos)

Adaptação

Aleatoriedade

Comunicação

Introdução

□ Diferentes técnicas/algoritmos

- Simples
- Meta-heurísticas (métodos estocásticos)

Adaptação

Aleatoriedade

Comunicação

Feedback

Introdução

□ Diferentes técnicas/algoritmos

- Simples
- Meta-heurísticas (métodos estocásticos)

Adaptação

Aleatoriedade

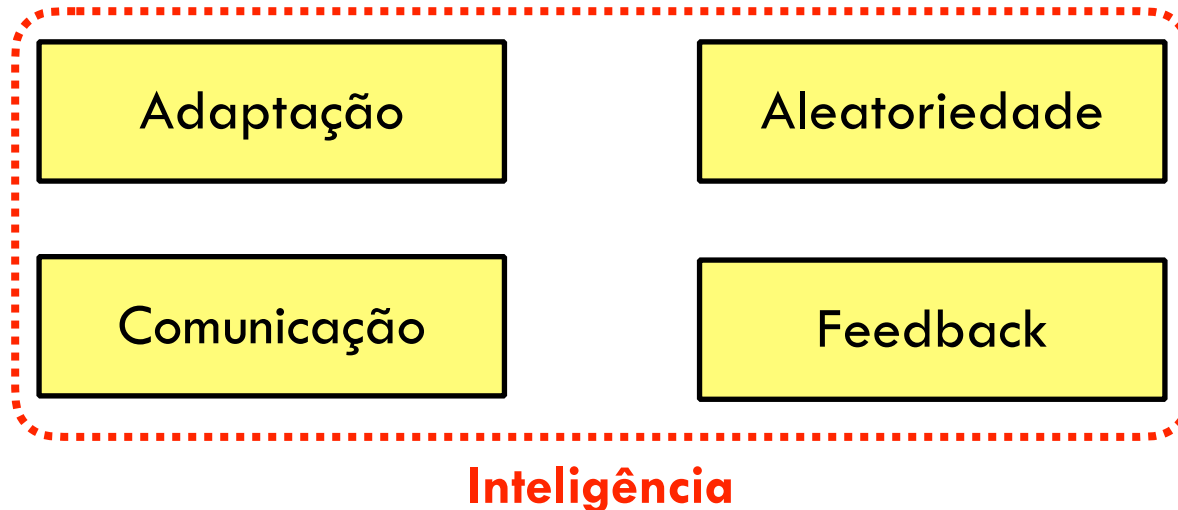
Comunicação

Feedback

Introdução

□ Diferentes técnicas/algoritmos

- Simples
- Meta-heurísticas (métodos estocásticos)



Introdução

- **Inteligência de Enxame (*Swarm Intelligence*)**



Roteiro



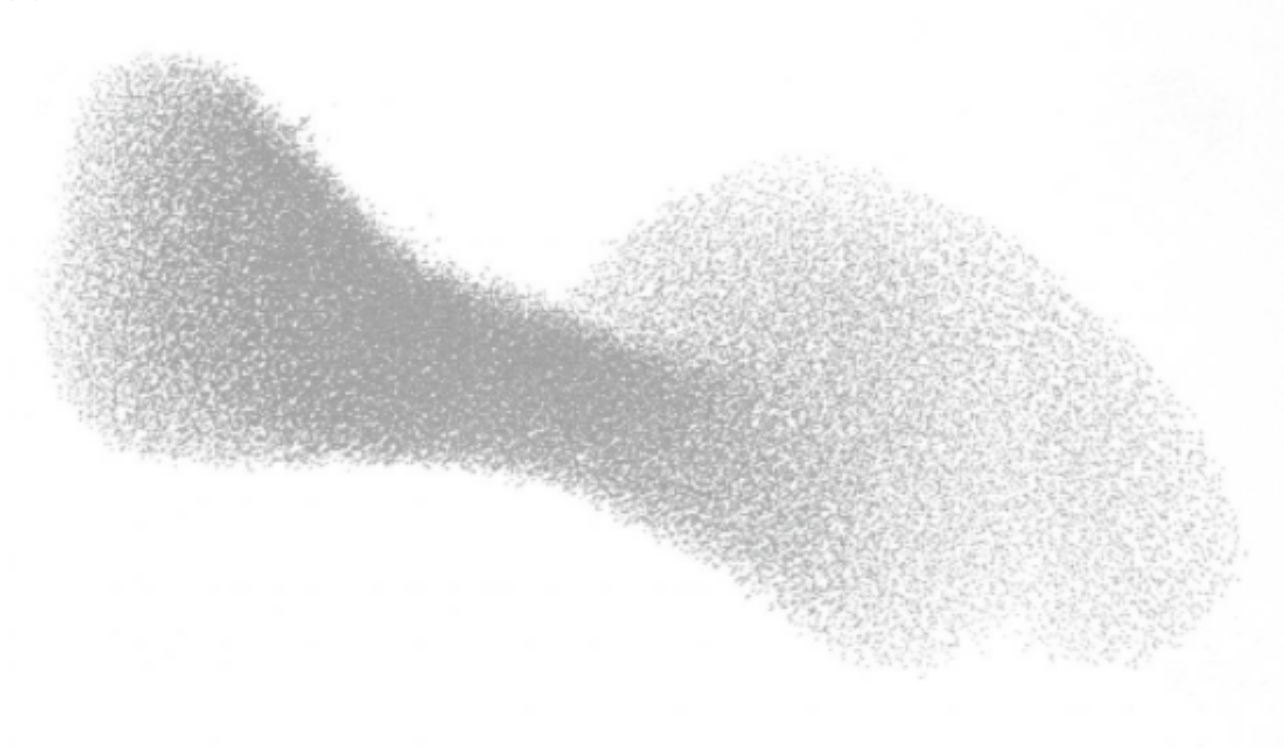
- 1 Introdução
- 2 *Swarm Intelligence*
- 3 *PSO*
- 4 Síntese / Próximas Aulas
- 5 Referências

Swarm Intelligence



Swarm Intelligence

- Uma coleção bem estruturada de **agentes** interagindo
- Agentes



Swarm Intelligence

- Uma coleção bem estruturada de **agentes** interagindo
- Agentes

- indivíduos que pertencem a um grupo
- comportamento guiado por regras simples (**inteligência limitada**)
- contribuem para, e em benefício do grupo
- Podem reconhecer, se comunicar e interagir um com o outro e com o ambiente

Swarm Intelligence



Colônia de formigas



Bando de pássaros



Cardume de peixes

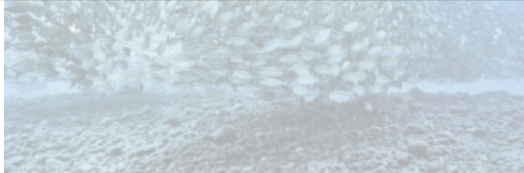


manada de zebras

Swarm Intelligence



- Estrutura dos agentes é:
 - **Descentralizada e auto-organizada**
- A ação de cada indivíduo mapeia as possibilidades para indicar o comportamento global



Cardume de peixes



manada de zebras

Swarm Intelligence

□ **Swarm Intelligence**

- tem origem na década de 90
- sistemas robóticos compostos por uma coleção de agentes simples
- **princípio:** agentes pouco inteligentes e com capacidade individual limitada, são capazes de apresentar comportamentos coletivos inteligentes

Swarm Intelligence

- **Características:**
 - **Proximidade:** agentes devem estar próximos o suficiente para poderem interagir
 - **Qualidade:** os agentes devem ter a capacidade de avaliar seus comportamentos
 - **Diversidade:** a organização dos agentes permite reagir a situações inesperadas
 - **Adaptabilidade:** deve conseguir se adequar as variações do ambiente

Swarm Intelligence

- Exemplos de Swarm Intelligence na literatura:
 - Enxame de partículas (*Particle Swarm Optimization*)
 - Colônia de formigas (*Ant Colony Optimization*)
 - Cardume de peixes (*Fish Swarm Intelligence*)
 - Enxame de vaga-lumes (*Firefly Algorithm*)
 - Colmeia de abelhas (*Bee Algorithm*)
 - ...

Swarm Intelligence

- Exemplos de Swarm Intelligence na literatura:
 - **Enxame de partículas (*Particle Swarm Optimization*)**
 - Colônia de formigas (*Ant Colony Optimization*)
 - Cardume de peixes (*Fish Swarm Intelligence*)
 - Enxame de vaga-lumes (*Firefly Algorithm*)
 - Colmeia de abelhas (*Bee Algorithm*)
 - ...

Roteiro



- 1 Introdução
- 2 *Swarm Intelligence*
- 3 *PSO*
- 4 Síntese / Próximas Aulas
- 5 Referências

PSO

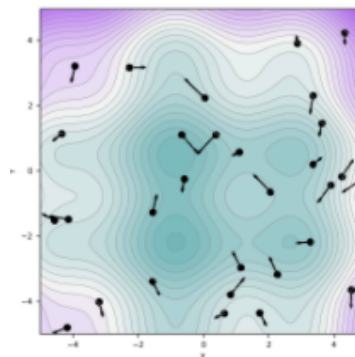
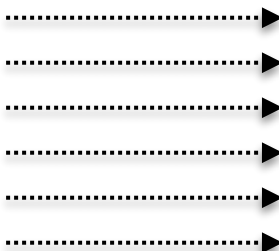
- Proposto por [Kennedy & Eberhart \(1995\)](#)
- Inspirado no inteligência coletiva de sistemas naturais
 - Método estocástico de otimização
 - Inteligência distribuída
 - Solução candidata = partícula
 - População = enxame
 - Melhorar:
 - desempenho coletivo
 - desempenho individual das partículas

PSO

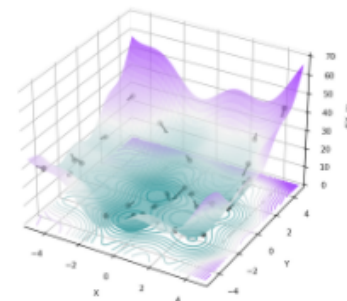
- Replicar comportamentos bioinspirados



Bando de pássaros



Enxame de Partículas

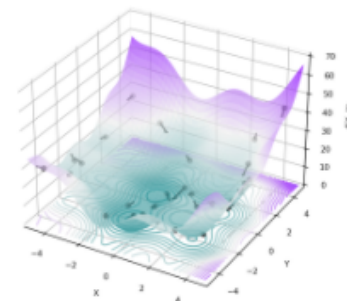
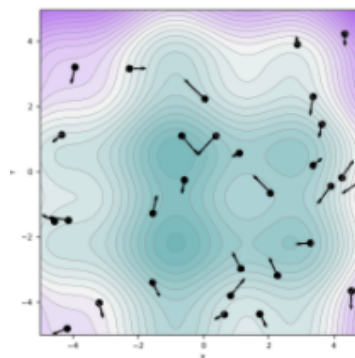
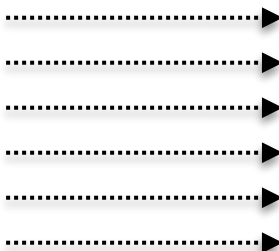


PSO

- Replicar comportamentos bioinspirados



Bando de pássaros

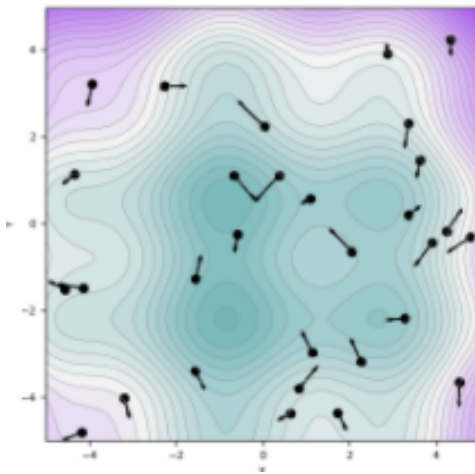


Enxame de Partículas

A base do algoritmo é a **interação** entre os agentes (**partículas**) da população movendo rumo a melhores posições do espaço de busca

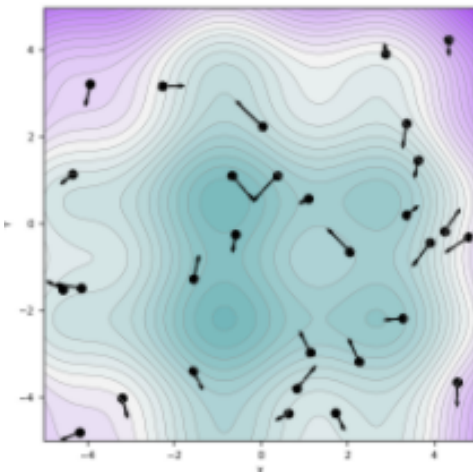
PSO

- Tem fundamentação também no comportamento humano:



PSO

- Tem fundamentação também no comportamento humano:

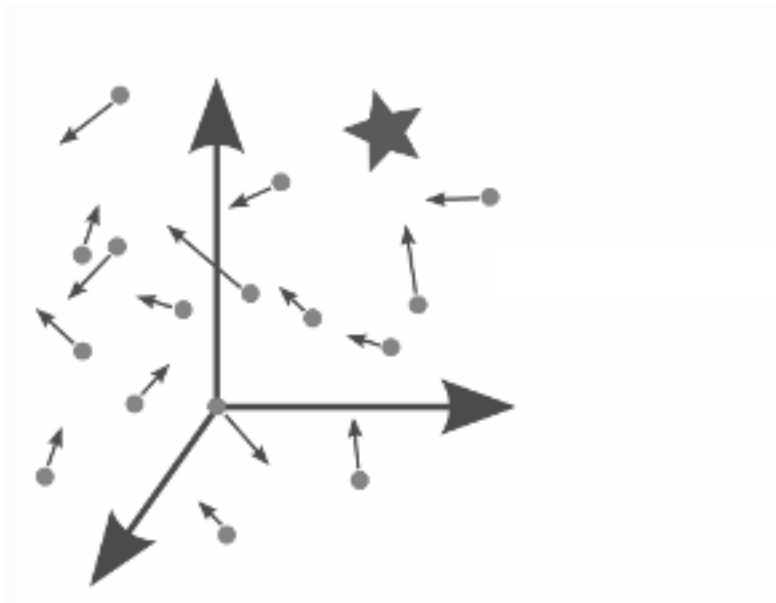


1. **Inércia:** tendemos a repetir fórmulas de sucesso que funcionaram no passado
2. **Influência da sociedade:** tentar emular o que outros conseguiram realizar com sucesso
3. **Influência de vizinhos:** aprendemos daqueles mais próximos. Modificamos nossos comportamentos com base nas histórias de sucesso/fracassos que ouvimos dos mais próximos

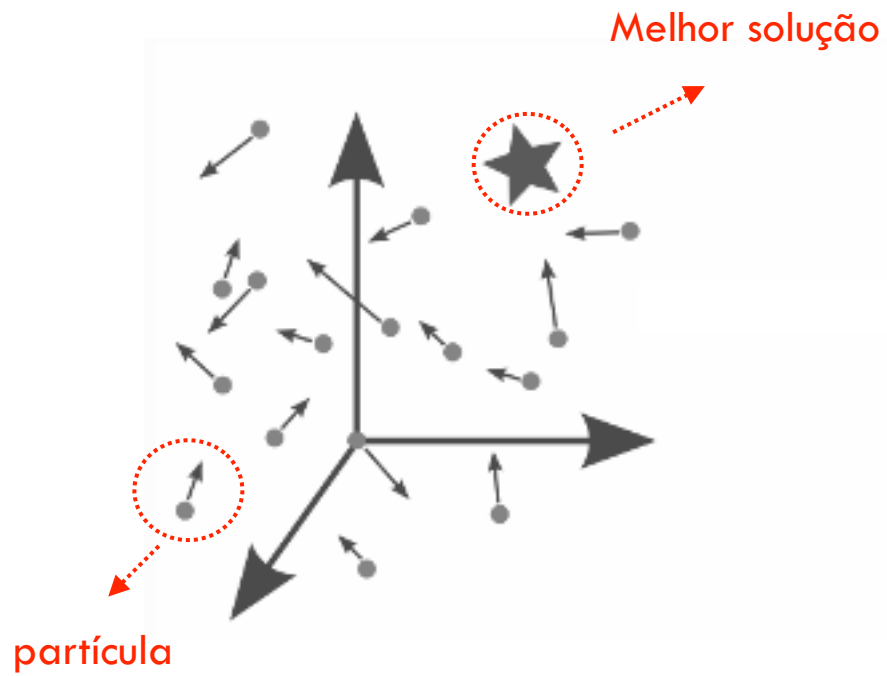
- **Teoria Sociocognitiva** explica o PSO:
 - cada agente possui sua **própria experiência** e é capaz de **avaliar a qualidade** dessa experiência
 - como os agentes são sociáveis, eles também possuem **conhecimento sobre** como **seus vizinhos** se comportam (desempenharam)
- Esses dois tipos de informação correspondem à aprendizagem individual (**cognitiva**) e à transmissão cultural (**social**), respectivamente.

- **Três princípios** são utilizados para resumir o processo de adaptação cultural/social:
 - 1 **Avaliação**: agentes/partículas possuem a capacidade de verificar o ambiente de forma a avaliar seu próprio comportamento
 - 2 **Comparação**: agentes usam uns aos outros como material comparativo
 - 3 **Imitação**: um dos centros da organização social humana, e é importante para a aquisição de manutenção das habilidades mentais

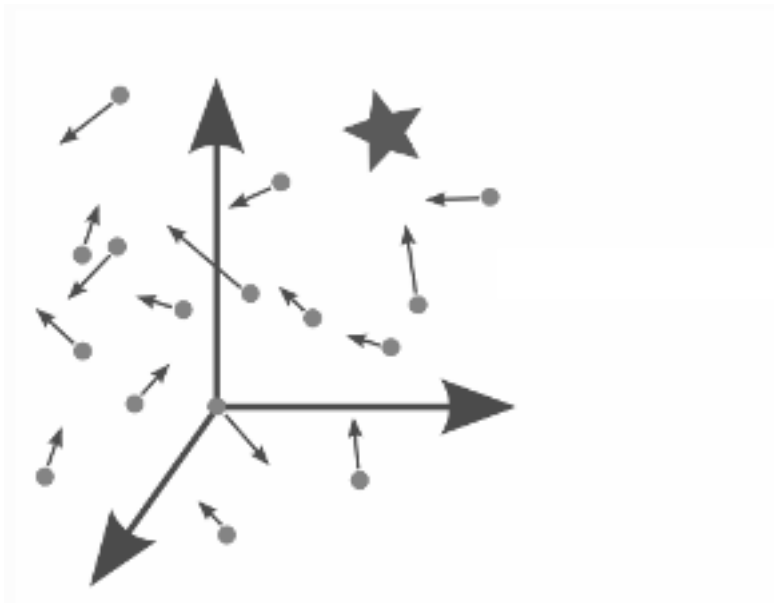
PSO



PSO

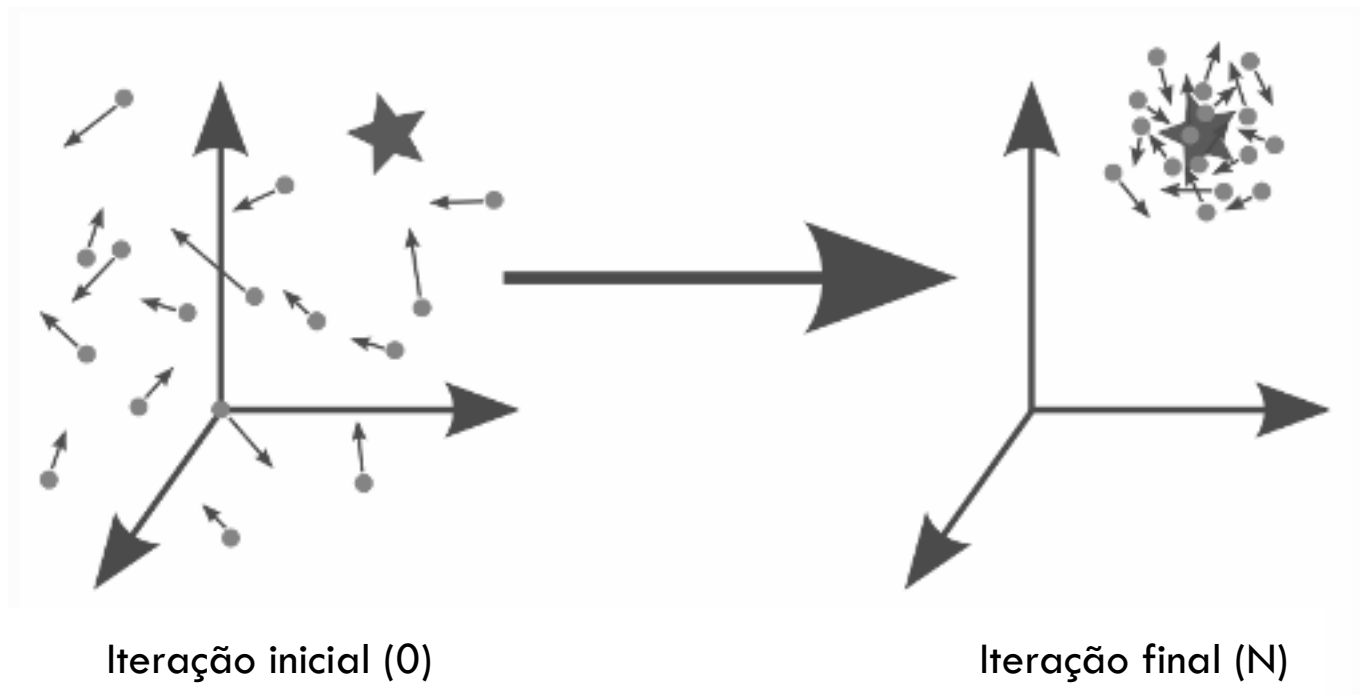


PSO

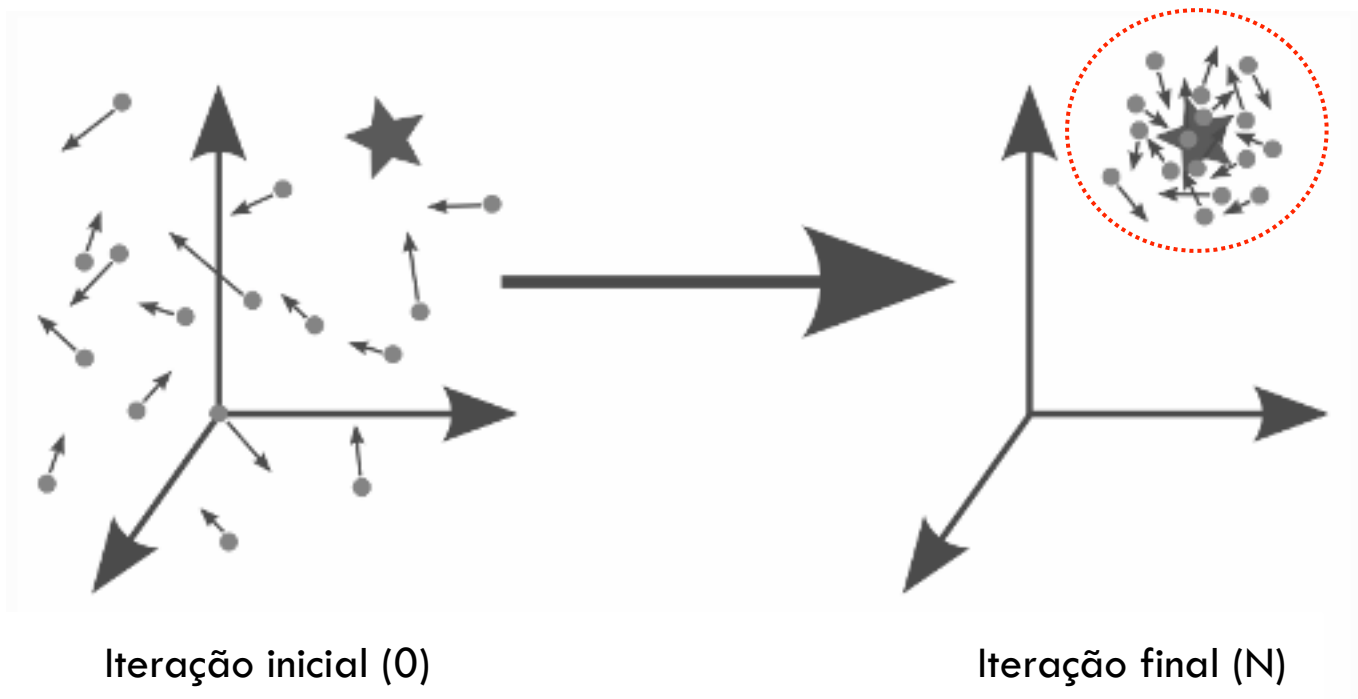


Iteração inicial (0)

PSO



PSO



A ideia é que essas partículas **tenderão** a se movem em direção umas das outras e vão influenciar umas as outras

PSO

□ Cada partícula: 

- **posição** da partícula no espaço (x)
- **velocidade** (v): direção e velocidade que a partícula viaja no espaço

PSO

□ Cada partícula: 

- **posição** da partícula no espaço (x)
- **velocidade** (v): direção e velocidade que a partícula viaja no espaço
- memória: **local best** → melhor posição atingida pela partícula desde sempre
- conhecimento: **global best** → melhor posição atingida pelo enxame como um todo

PSO

□ Cada partícula: 

- **posição** da partícula no espaço (x)
- **velocidade** (v): direção e velocidade que a partícula viaja no espaço
- memória: **local best** → melhor posição atingida pela partícula desde sempre
- conhecimento: **global best** → melhor posição atingida pelo enxame como um todo
- vizinhança: **neighbors best** → melhor posição atingida por N vizinhos

PSO

□ Cada partícula: ●→

- A posição de uma partícula i em um instante de tempo é dada por:

$$x_i(t + 1) = x_i(t) + v_i(t + 1)$$

A posição futura de uma partícula depende de sua posição anterior e da velocidade da partícula

PSO

□ Cada partícula: 

- A posição de uma partícula i em um instante de tempo é dada por:

$$x_i(t + 1) = x_i(t) + v_i(t + 1)$$

$$v_i(t + 1) = v_i(t) + c_1(P_i^{best}(t) - x_i(t)) + c_2(P_{gbest} - x_i(t))$$

PSO

- - **c1** (individual) e **c2** (grupo) são os coeficientes de aceleração (taxa de aprendizado)
 - **Pi_best** é a melhor posição conhecida pela partícula i
 - **Pgbest** é a melhor posição conhecida pelo enxame

$$x_i(t + 1) = x_i(t) + v_i(t + 1)$$

$$v_i(t + 1) = v_i(t) + c_1(P_i^{best}(t) - x_i(t)) + c_2(P_{gbest} - x_i(t))$$

PSO

- Cada partícula: 

$$\mathbf{x}_i(t + 1) = \mathbf{x}_i(t) + \mathbf{v}_i(t + 1)$$

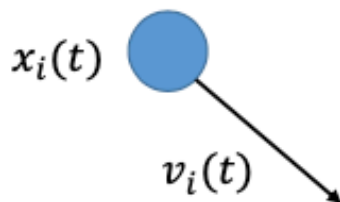
$$\mathbf{v}_i(t + 1) = \boxed{\mathbf{v}_i(t)} + \boxed{c_1(P_i^{best}(t) - \mathbf{x}_i(t))} + \boxed{c_2(P_{gbest} - \mathbf{x}_i(t))}$$

Fator de inércia Componente cognitivo Componente social

PSO

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + c_1(P_i^{best}(t) - \mathbf{x}_i(t)) + c_2(P_{gbest} - \mathbf{x}_i(t))$$

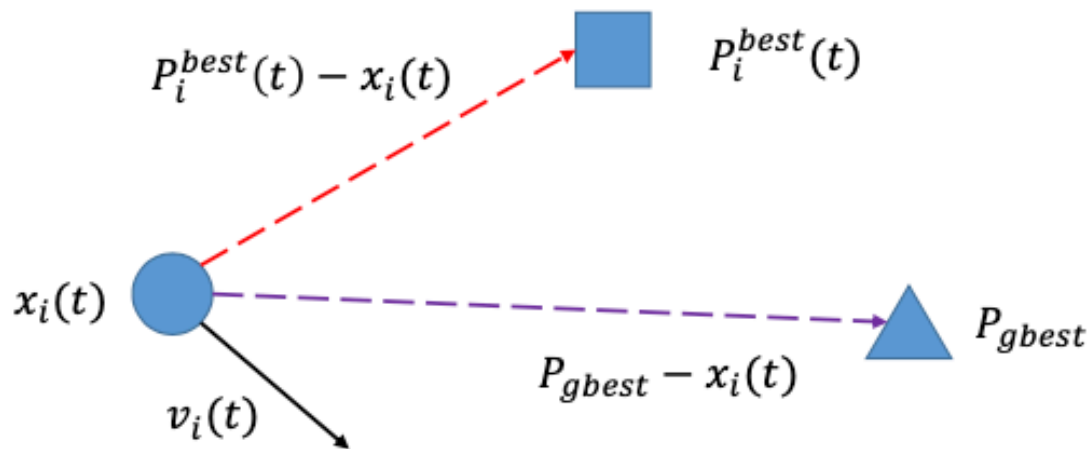
■ $P_i^{best}(t)$



▲ P_{gbest}

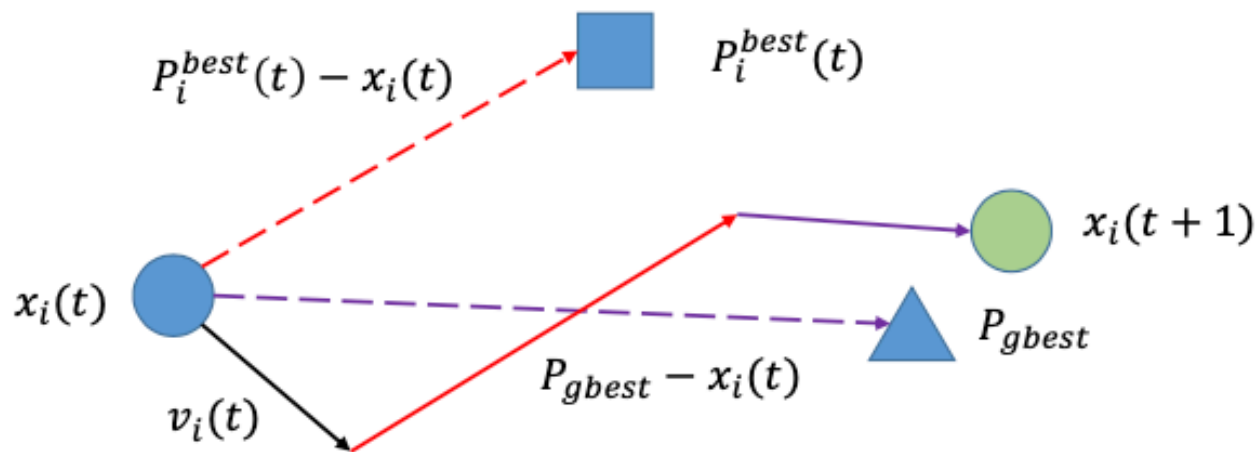
PSO

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + c_1(P_i^{best}(t) - \mathbf{x}_i(t)) + c_2(P_{gbest} - \mathbf{x}_i(t))$$



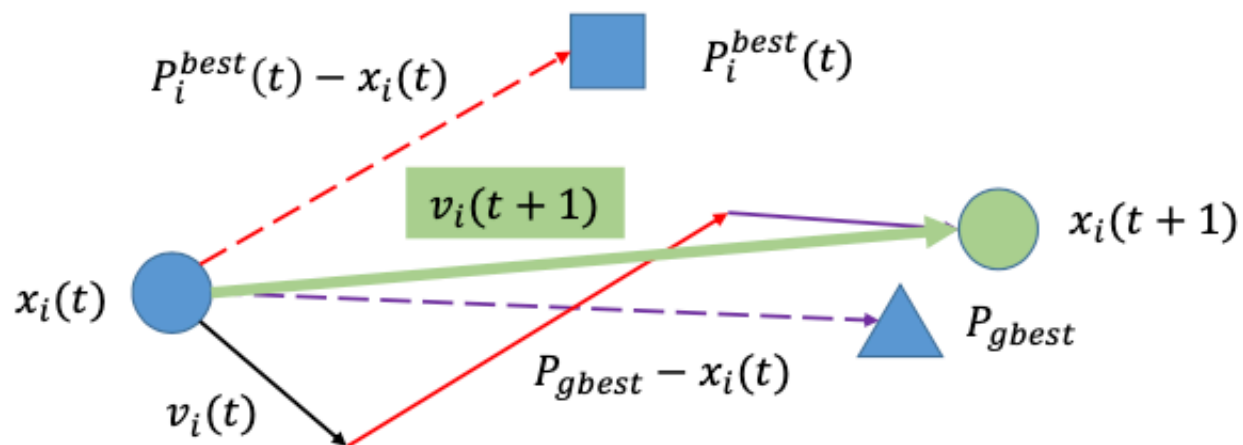
PSO

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + c_1(P_i^{best}(t) - \mathbf{x}_i(t)) + c_2(P_{gbest} - \mathbf{x}_i(t))$$



PSO

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + c_1(p_i^{best}(t) - \mathbf{x}_i(t)) + c_2(P_{gbest} - \mathbf{x}_i(t))$$



Algoritmo PSO

Entradas: $\min f(x)$, onde $A \leq x \leq B$

- **N:** número de partículas do enxame
- **n.iter:** número máximo de iterações

Início algoritmo

1. Criar uma população inicial (P) de N indivíduos (x_i) com valores aleatórios entre A e B
 - o ideal é que a população seja distribuída por todo o espaço de busca
 - a partícula é denotada por um vetor de coordenadas (similar ao cromossomo do GA)

Algoritmo PSO

2. Inicializar o vetor de velocidade (v_i) de cada partícula (x_i)
 - aleatoriamente ou serem nulos (zero)
3. Inicializar as melhores posições (***localbest i***) já encontrada para cada partícula (x_i)
4. Inicializar a melhor posição já encontrada pelo enxame (***globalbest***)
5. Enquanto critério de parada não for satisfeito
 - 5.1. Avaliar cada partícula x_i em P ($f(x_i)$)
 - 5.2 Atualizar melhores posições locais (***localbest i***)
 - 5.3 Atualizar melhor posição global (***globalbest***)
 - 5.4 Atualizar os vetores de velocidade de cada partícula (v_i)
 - 5.5. Atualizar as posições de cada partícula (x_i)
6. Retornar (***globalbest***)

PSO

□ Algoritmo

```
1  Gerar a população inicial aleatoriamente
2
3  while(condição de parada não satisfeita)
4      for i = 1 to size(população)
5          if  $f(x_i) < f(p_i^{best})$ 
6               $p_i^{best} = x_i$  //melhor performance individual até o momento
7          if  $f(p_i^{best}) < f(P_{gbest})$  //se  $x_i$  alcançou a melhor solução da população
8               $P_{gbest} = x_i$ 
9          for d = 1 to L //L é a dimensão do problema
10             atualiza_velocidade() //equação v(i)
11             atualiza_posição() //equação x(i)
```

- Algumas modificações do algoritmo PSO alteram as linhas 7 e 8. Ao invés de atualizar o valor do Pgbest do enxame todo, modificam o melhor valor dos k vizinhos da partícula i

<i>Característica</i>	<i>Descrição</i>
Indivíduo (agente)	Partícula
População de indivíduos	Enxame de partículas
Esquecimento e aprendizagem	Incremento ou decremento nos valores de alguns atributos das partículas
Experiência própria de um indivíduo	Cada partícula possui conhecimento de sua história (desempenho) e usa este conhecimento para direcionar seus próximos movimentos no espaço
Interação social	Cada partícula também possui conhecimento sobre o desempenho de outras partículas e emprega de igual maneira esse conhecimento para as suas próximas escolhas

PSO

□ **Vantagens:**

- Versatilidade
- Fácil implementação
- Custo computacional (paralelizável)
- Não existem muitos parâmetros para configurar

PSO

□ **Contras**

- pode ficar preso em ótimos locais (como resolver?)
- Uma das tentativas é deixar o funcionamento do algoritmo mais estocástico, propiciando a fuga dos ótimos locais
 - Atribuir pesos ao fato de inércia, componente cognitivo e social

$$v_i(t + 1) = wv_i(t) + r_1c_1(P_i^{best}(t) - x_i(t)) + r_2c_2(P_{gbest} - x_i(t))$$

- Em que w , r_1 e r_2 são valores aleatórios reais no intervalo $[0,1]$

Hands on



Vamos exercitar :)
[Google Colab - [Exemplo Tuning](#)]

Roteiro

- 1 Introdução
- 2 *Swarm Intelligence*
- 3 *PSO*
- 4 **Síntese / Próximas Aulas**
- 5 Referências

Síntese

□ Inspiração Biológica

- Comportamento coletivo
- Cardumes / Enxames / Populações de animais
- Inteligência de Enxame
- *Particle Swarm Optimization (PSO)*

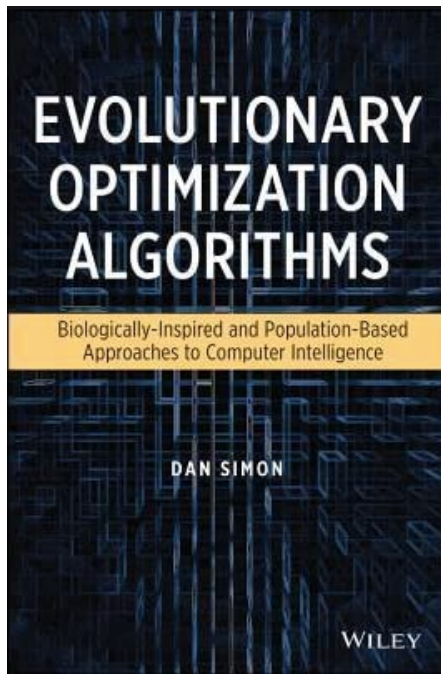
Próxima aula

- Fluxo de Ciência de Dados
 - Etapas
 - Relação com conteúdos já estudados
 - Exercícios/Exemplos

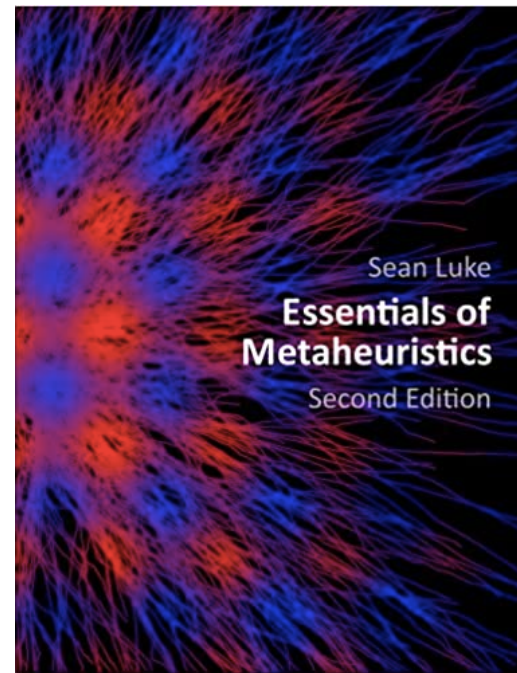
Roteiro

- 1 Introdução
- 2 *Swarm Intelligence*
- 3 *PSO*
- 4 Síntese / Próximas Aulas
- 5 Referências

Referências



[Simon, 2013]



[Luke, 2013]

Referências



[Simon 2013] Dan Simon. Evolutionary Optimization Algorithms: Biologically inspired and population-based approaches to computer intelligence. Wiley, 2013.

[Luke 2013] Sean Luke. Essentials of Metaheuristics, Lulu, second edition, 2013. Available at <http://cs.gmu.edu/~sean/book/metaheuristics/>



Obrigado :)

Rafael G. Mantovani

rafaelmantovani@utfpr.edu.br

Material Complementar



Vídeos

- <https://www.youtube.com/watch?v=G1t4M2Xnlhl>
- <https://www.youtube.com/watch?v=Wp3Gau-AIjs>

Links:

- <https://github.com/7ossam81/EvoloPy>