

1ª. AVALIAÇÃO PRÁTICA

DISCIPLINA	MACHINE LEARNING			DATA	
PROFESSOR	LINDENBERG ANDRADE			TIPO DE AVALIAÇÃO	ATIVIDADE I
TURMA	4NA	CÓDIGO DA TURMA	NLN0400104NNA	NOTA	

AVALIAÇÃO PRÁTICA I (2 PONTOS)

Descrição da Atividade – Aplicação de Algoritmos Metaheurísticos

Para esta atividade, a partir do **tema sorteado**, é necessário aplicar **Algoritmos Metaheurísticos** para resolver problemas reais, em situações onde suas técnicas se mostram mais eficazes.

Finalidade do trabalho:

O objetivo é demonstrar, de forma prática, a aplicação dos Algoritmos Metaheurísticos, permitindo que os discentes compreendam não apenas os conceitos teóricos, mas também sua implementação em situações reais.

O trabalho busca:

- **Explorar a capacidade dos algoritmos Metaheurísticos:** Resolver problemas complexos que são difíceis de serem solucionados por métodos tradicionais.
- **Desenvolver habilidades práticas:** Programar, testar e integrar soluções computacionais dentro do grupo, reforçando a colaboração e a divisão de funções.
- **Analisar o desempenho das soluções:** Comparar resultados, identificar melhorias e interpretar os resultados obtidos.
- **Relacionar teoria e prática:** Aplicar conceitos de otimização, busca local e global, e estratégias adaptativas em cenários reais.
- **Estimular pensamento crítico e criatividade:** Propor soluções alternativas, ajustar parâmetros e avaliar diferentes abordagens para o problema.

1ª. AVALIAÇÃO PRÁTICA

Ao final, espera-se que o grupo entregue uma solução funcional, bem documentada e demonstrando **compreensão teórica e prática**, além da capacidade de trabalhar de forma colaborativa.

REGRAS E DISTRIBUIÇÃO DE FUNÇÕES

- Esta avaliação prática valerá 2 pontos que somará a nota da Avaliação 1 da disciplina de Machine Learning.
- A atividade deverá ser realizada por um grupo de até 5 alunos, cada um com uma função específica.
- Cada grupo deverá apresentar o trabalho em sala de aula no dia definido pelo docente, e enviar um relatório para o professor via e-mail ou formulário disponibilizado pelo mesmo.
- Todos os membros deverão participar ativamente de todas as etapas da atividade, pois a colaboração integral será considerada na avaliação.
- Caso o grupo não preencha as 5 vagas, a função de **auxiliar** não será necessária, e todos os membros deverão desempenhar coletivamente as tarefas correspondentes.
- O relatório deve ser escrito em LaTeX (<https://pt.overleaf.com/>) com o modelo disponibilizado pelo docente. O relatório deve conter a explicação da atividade, destacando os motivos de escolha da aplicação do tema, deve incluir o código em apêndice e o link da programação disponibilizado no GitHub (<https://github.com/>).
- Toda a Programação feita deverá ser implementada em Python, fazendo uso do Google Colab (<https://colab.google/>) que deverá ser salva no GitHub (O GitHub permite que o Colab salve arquivos no repositório).
- A apresentação deverá ser feita por meio de slides, que deverão ser enviados até 3 horas antes do início da aula, para que não ocorram atrasos nas apresentações. O envio poderá ser feito pelo canal de envio disponibilizado pelo docente (e-mail ou WhatsApp).
- O trabalho deverá ser apresentado no dia e horário definido pelo docente e cada grupo terá o tempo 10-15 minutos para realizar a apresentação e mais 5 minutos para responder perguntas do professor e da turma.

1ª. AVALIAÇÃO PRÁTICA

FUNÇÕES:

- **Líder (1 aluno):** Organiza o grupo, distribui tarefas, garante o cumprimento dos prazos e coordena a comunicação entre os membros. Também deve acompanhar o progresso de cada função e apoiar colegas que encontrem dificuldades.
- **Redator (1 aluno):** Responsável por escrever e revisar os textos do trabalho, garantindo clareza, coesão e correção da linguagem. Os demais colegas devem colaborar com informações, revisões e sugestões.
- **Programador 1 (1 aluno):** Principal responsável pelo desenvolvimento do código. Trabalha em colaboração com o grupo para implementar as funcionalidades planejadas, garantindo que a parte desenvolvida funcione corretamente e esteja integrada ao projeto.
- **Programador 2 (1 aluno):** Foca nos testes e aperfeiçoamento da apresentação dos resultados. Complementa o trabalho do Programador 1, ajudando na integração final e realizando ajustes necessários para garantir que o código funcione plenamente. Os demais colegas do grupo devem auxiliar nesse processo.
- **Auxiliar (1 aluno):** Dá suporte a todas as funções, auxiliando na pesquisa, execução de testes, elaboração do relatório e na resolução de problemas que surgirem durante a atividade.

TEMAS:

1. Algoritmos Meméticos
2. Evolução Diferencial
3. Programação Genética
4. Otimização por Enxame de Partículas
5. Algoritmo de Abelhas
6. Algoritmo de Vagalumes
7. Recozimento Simulado
8. Busca Tabu
9. Busca Harmônica
10. Otimização por Tempestade de Raios
11. Busca Local Iterada GRASP

1ª. AVALIAÇÃO PRÁTICA

12. Busca de Vizinhaça Variável

13. Algoritmos Multiarranque