

CASE – BOLSISTA GRADUADO – APRENDIZADO DE MÁQUINA - 2025

Nome Completo: _____

Data: ____/____/____ Hora do início do teste: _____ Hora do final do teste: _____

INTRODUÇÃO

Seja bem-vindo ao teste para candidatos à vaga de Bolsista em Inteligência Artificial (Aprendizado de Máquina) – RP 16497, 16499 e 16500 – SENAI Londrina/PR. Por favor, registre seu nome, o horário de início e o horário de término do teste. O prazo para entrega é de 32 horas a partir do início da atividade. Portanto, leia atentamente o objetivo e o briefing, e organize seu ambiente de trabalho para realizar as tarefas de maneira tranquila e focada.

A análise e interpretação dos requisitos descritos nesta atividade também serão avaliadas, e todo o esforço demonstrado será considerado no processo seletivo.

Ao concluir o teste, envie um e-mail para rafael.bressan@sistemapiep.org.br com o título "CASE – Bolsista - RP 16497, 16499, 16500", relatando como foi sua experiência na execução da atividade, destacando os principais desafios e pontos de maior complexidade encontrados durante o desenvolvimento.

Ressaltamos que todas as informações contidas neste documento devem ser mantidas em sigilo, respeitando os direitos de autoria e a proteção das informações aqui apresentadas.

OBJETIVO

O objetivo deste teste é avaliar seu **conhecimento técnico, capacidade de síntese, criatividade e organização** no desenvolvimento de sistemas baseados em aprendizado de máquina. A proposta está fundamentada nos requisitos mínimos necessários para a concepção de sistemas computacionais **adaptáveis a diferentes cenários e necessidades**.

Além da execução das atividades, também será considerada a sua **capacidade de aplicar conhecimentos já adquiridos e de evoluir em conteúdos novos ou menos dominados**, valorizando não apenas a conclusão das tarefas, mas também a qualidade, o raciocínio e a evolução demonstrada ao longo do desenvolvimento.

BRIEFING

Case – Eficiência Energética

Uma empresa está implementando em seu ambiente de produção um sistema preditivo voltado para eficiência energética de edifícios. O objetivo é prever o consumo de energia para aquecimento e resfriamento com base em características arquitetônicas e construtivas dos imóveis.

Para essa PoC (Proof of Concept), propomos a criação de um sistema de regressão supervisionada utilizando o conjunto de dados público **UCI Energy Efficiency Dataset**, disponível no link: <https://archive.ics.uci.edu/ml/datasets/energy+efficiency>

Descrição do conjunto de dados:

O dataset contém 768 amostras de edifícios, com as seguintes características:

- Variáveis de entrada:
 - Relative Compactness
 - Surface Area
 - Wall Area
 - Roof Area
 - Overall Height
 - Orientation
 - Glazing Area
 - Glazing Area Distribution
- Variáveis de saída (dupla saída):
 - Heating Load (Carga de aquecimento)
 - Cooling Load (Carga de resfriamento)

Nesse contexto, realize as atividades listadas a seguir. Como sugestão, inicie pela atividade com a qual você possui maior familiaridade e, após concluí-la, avance para as demais.

Atividades a serem desenvolvidas:

♦ 1. AT1 - Análise exploratória dos dados

- Realize uma análise detalhada da distribuição dos dados e elabore um relatório (em formato PDF) apresentando suas considerações.
- **Entrega:**
 - Código-fonte utilizado para a análise.
 - PDF contendo a análise realizada.

♦ 2. AT2 - Modelagem e avaliação de algoritmos

- Investigue e aplique pelo menos **cinco técnicas de aprendizado supervisionado** de regressão para o problema proposto.
- Avalie o desempenho de cada técnica utilizando métricas apropriadas (por exemplo: RMSE, MAE, R^2).
- Também investigue e aplique estratégias de **pré-processamento dos dados**, tais como normalização, padronização, tratamento de outliers ou seleção de features.
- **Entrega:**
 - O **código** de todos os experimentos desenvolvidos.
 - Um **relatório em PDF** contendo a análise comparativa entre os modelos testados.

♦ 3. AT3 - Desenvolvimento de um sistema web para predição

- Desenvolva uma aplicação web em **Python (Django)** que contenha:
 - **Sistema de login.**
 - O sistema deverá permitir que o usuário realize login na aplicação utilizando suas credenciais de acesso previamente cadastradas. A autenticação deve garantir segurança e controle de acesso à aplicação.

- Usuário padrão para testes:
 - E-mail: admin@admin.com
 - Password: Senai@2025
- **Gerenciamento de usuários**
 - O sistema deverá permitir que o usuário logado seja capaz de cadastrar novos usuários, consultar os dados de um usuário específico, listar todos os usuários registrados, bem como realizar a edição e exclusão de usuários existentes.
 - Dados obrigatórios para cada usuário:
 - Nome completo
 - E-mail
 - Telefone
 - Senha de acesso
- **Área para testar o modelo de regressão:**
 - O sistema deverá disponibilizar uma interface onde o usuário possa **informar os valores das features de entrada** correspondentes aos atributos utilizados no treinamento do modelo.
 - Após o envio dos dados, o sistema deve **processar a requisição e retornar as predições** geradas pelo modelo de regressão, apresentando os valores estimados de **carga de aquecimento** e **carga de resfriamento** para o cenário informado.
 - O modelo utilizado deverá ser aquele que **obteve o melhor desempenho na etapa anterior de experimentação** e deverá ser **previamente treinado e carregado de forma separada da aplicação**, garantindo assim a separação entre os processos de treinamento e de inferência.
 - Essa funcionalidade deve estar protegida por autenticação e integrada ao modelo final.
- **Banco de Dados:**
 - Todas as entradas fornecidas pelos usuários, bem como os resultados das predições geradas pelo modelo, deverão ser **armazenadas em um banco de dados PostgreSQL**, garantindo a persistência, integridade e rastreabilidade das informações.
- **Funcionalidades adicionais:**
 - **Visualizar todas as predições realizadas**, permitindo a consulta dos dados de entrada informados e dos respectivos resultados gerados.
 - **Excluir uma predição registrada**, possibilitando a remoção de registros específicos do banco de dados conforme a necessidade.
- **Liberdade de escolha tecnológica para o front-end:**
 - O candidato terá total liberdade para escolher o melhor layout, template e tecnologias de front-end que considerar mais adequados para a aplicação. A interface pode ser desenvolvida com ferramentas como HTML/CSS puro, frameworks modernos (como React, Vue ou Angular), ou ainda com templates prontos, desde que estejam devidamente integrados à aplicação e descritos no README.
- **Entrega:**
 - O código-fonte deverá ser acompanhado de um arquivo **README.md** contendo **instruções detalhadas para a execução do projeto**, incluindo orientações sobre instalação de dependências, configuração do ambiente e procedimentos para inicialização da aplicação.

◆ 4. Observações

- Todo o projeto, incluindo os códigos-fonte, scripts, documentação e relatórios, deverá ser disponibilizado em um **repositório no GitHub**, seguindo as boas práticas de versionamento baseadas no **Gitflow**.

- O link para acesso ao repositório deverá ser enviado por e-mail, conforme as orientações previamente estabelecidas.
- A codificação deve respeitar as **boas práticas da linguagem Python**, preferencialmente utilizando a versão **Python 3.10** e adotando o **paradigma de programação orientada a objetos (POO)**.
- A atividade **AT3 – Desenvolvimento do sistema web para predição** deverá ser, preferencialmente, empacotada utilizando **Docker**, com suporte a **docker-compose** para facilitar a criação e a execução do ambiente.
- **Critérios adicionais para avaliação das entregas:**
 - Estrutura e organização do repositório no GitHub (ex.: separação clara entre código-fonte, documentação e modelos treinados).
 - Qualidade do código (clareza, legibilidade, organização em classes e funções, modularidade e reuso de código).
 - Uso correto de versionamento e de convenções no Git (commits claros e organizados).
 - Funcionalidade completa e funcionamento adequado das aplicações entregues.
 - Clareza e completude do **README.md** (instruções de instalação, execução, dependências, ambiente Docker, explicação sobre o funcionamento da solução).
 - Coerência, profundidade e qualidade dos relatórios entregues em PDF (análise de dados, comparação de modelos e relato de dificuldades).
 - Uso adequado de práticas de desenvolvimento seguro e boas práticas de integração entre componentes (API, Banco de Dados, Modelos de IA).

Importante:

O prazo para entrega será de **32 horas** a partir do recebimento deste case, ou seja, contado a partir do envio deste material por e-mail.

Todo o material (códigos (compactados em formato .zip), relatórios em PDF e link para o repositório no GitHub) deverá ser enviado por e-mail conforme as orientações previamente disponibilizadas.