

Olimpíadas Brasileiras Online 2026

CONTEÚDO OBOIA

I Disposições Preliminares

- i) O presente documento estabelece quais conteúdos teóricos podem ser cobrados nas provas da Olimpíada Brasileira Online de Inteligência Artificial (OBOIA).
- ii) A OBOIA será realizada em duas fases.

II Nível Aberto

O conteúdo programático da OBOIA, válido para ambas as fases, é o listado abaixo. Tópicos não listados são considerados excluídos e não serão cobrados. A versão mais atualizada desta lista poderá ser encontrada na página oficial da OBOIA <https://noic.com.br/obo/ia>.

i) Programação e ferramentas.

- (a) Python: laços, funções e estruturas de dados;
- (b) NumPy e pandas para manipulação de dados;
- (c) Matplotlib e Seaborn para visualização;
- (d) scikit-learn para tarefas clássicas de aprendizado de máquina;
- (e) PyTorch: criação e manipulação de tensores (arranjos multidimensionais);
- (f) execução e treinamento de modelos em CPU e GPU.

ii) Tratamento de dados e engenharia de atributos.

- (a) Dados faltantes e irregulares: imputação por média, mediana e preenchimento para frente em séries;
- (b) preenchimento (*padding*) de sequências de comprimento variável;
- (c) normalização e padronização;
- (d) divisão dos dados em treino, validação e teste;
- (e) aumento de dados básico: espelhamento, recorte e adição de ruído;
- (f) tokenização e construção de vocabulário para texto e áudio, e divisão de imagens em *patches*;
- (g) codificação de variáveis categóricas (*one-hot*);
- (h) extração de atributos a partir de dados brutos, de alta dimensionalidade ou de comprimento variável: janelas deslizantes, operações de *pooling*, estatísticas resumo (média, desvio padrão) e representações obtidas por PCA ou por redes neurais.

iii) Aprendizado supervisionado.

- (a) Regressão linear e regressão logística;

- (b) regularização L1 e L2;
- (c) K-vizinhos mais próximos (K-NN);
- (d) árvores de decisão;
- (e) máquinas de vetores de suporte (SVM);
- (f) métodos de *ensemble*: *bagging*, *random forest* e *gradient boosting*.

iv) Aprendizado não supervisionado.

- (a) K-médias;
- (b) análise de componentes principais (PCA);
- (c) redução de dimensionalidade para visualização: t-SNE e UMAP;
- (d) agrupamento por densidade, hierárquico e espectral: DBSCAN, agrupamento hierárquico e *spectral clustering*.

v) Avaliação de modelos e metodologia.

- (a) Métricas de classificação e de regressão: acurácia, precisão, revocação, F1 e afins;
- (b) matriz de confusão e curvas ROC;
- (c) subajuste e sobreajuste;
- (d) validação cruzada;
- (e) ajuste de hiperparâmetros.

vi) Redes neurais: fundamentos.

- (a) Perceptron;
- (b) funções de ativação: ReLU, sigmoide e tangente hiperbólica;
- (c) funções de perda: MSE, MAE, entropia cruzada e afins;
- (d) gradiente descendente;
- (e) retropropagação.

vii) Treinamento de redes profundas.

- (a) Perceptron multicamadas (MLP);
- (b) gradiente descendente estocástico e por minilotes;
- (c) otimizadores com momento: Adam e AdamW;
- (d) taxa de aprendizado e comportamento de convergência;
- (e) inicialização de pesos;
- (f) normalização em lote (*batch normalization*);
- (g) regularização: *dropout*, parada antecipada e decaimento de peso;
- (h) ajuste fino de modelos pré-treinados, completo e com métodos eficientes em parâmetros.

viii) Arquiteturas e blocos fundamentais.

- (a) Representações vetoriais (*embeddings*) de texto, imagem e áudio;

- (b) operações de *pooling*: máximo e média;
- (c) mecanismo de atenção;
- (d) *transformers*, aplicados a texto e a imagem;
- (e) autocodificadores (*autoencoders*).

ix) Visão computacional.

- (a) Camadas convolucionais;
- (b) classificação de imagens;
- (c) detecção de objetos: YOLO, SSD e DETR;
- (d) segmentação de imagens: U-Net;
- (e) codificadores visuais pré-treinados, como ResNet;
- (f) aumento de dados para imagens;
- (g) aprendizado autossupervisionado em visão;
- (h) codificadores imagem-texto, como CLIP;
- (i) geração de imagens com GANs;
- (j) modelos de difusão.

x) Processamento de linguagem natural.

- (a) Classificação de texto;
- (b) codificadores de texto pré-treinados, como BERT;
- (c) modelagem de linguagem;
- (d) modelos codificador-decodificador, aplicados por exemplo a tradução automática e a modelagem visão-linguagem;
- (e) modelos de linguagem pré-treinados, tanto de código aberto quanto acessados por API.

xi) Áudio.

- (a) Codificadores de áudio pré-treinados, como HuBERT;
- (b) modelos de áudio, como Whisper, Qwen-Audio e Voxtral.

Os dados tratados nas provas poderão ser textuais, tabulares, de imagem, de áudio, de vídeo ou séries temporais, e deverão ser processados com os métodos acima.

III Contato

- i) Acompanhe nosso Instagram [@projetonoi](#) e participe de nossas comunidades no [WhatsApp](#) para obter atualizações sobre as OBOs.
- ii) Eventuais questionamentos sobre as OBOs devem ser encaminhados para o e-mail obonline@noic.com.br.