

Monstrinhos de Redes Neurais (NN)

1 "Eu podia jurar que não veria grafos nunca mais na minha vida" — aluno da turma 2023 que não quis se identificar

Objetivo: faça em uma folha de papel o grafo computacional da expressão abaixo. Realize o forward pass considerando que L é o vértice folha e $y = -50$, $a = -2$, $b = 3$, $c = 8$, $d = -5$, $e = 4$, $f = 1$ e $g = -10$. Compute os gradientes locais de cada vértice numérico realizando o backpropagation a partir do vértice folha. Indique para cada vértice qual é a respectiva derivada parcial e como a regra da cadeia é aplicada (faça de maneira similar ao que fizemos em sala).

$$L = (y - (a \cdot b + c \cdot d + e \cdot f + g))^2$$

Comentário: é necessário que a entrega seja em folha de papel mesmo, não é para fazer digital.

2 "Átomos não são bolinhas e ligações químicas não são pauzinhos" — Prof. Julio

Objetivo: Utilize classes de Python para modelar elementos químicos e moléculas.

Considerações do experimento: Crie uma classe chamada `Elemento`. Todo elemento químico utilizado nesta tarefa deve ser uma instância desta classe e deve pelo menos conter o símbolo do elemento, o seu número atômico e seu peso atômico como atributos (fique à vontade para incluir outras informações se quiser). Crie uma classe chamada `Molecula`. Esta classe deve receber uma relação de elementos químicos e suas quantidades, e deve ser capaz de exibir o peso atômico da molécula criada e de gerar uma fórmula química (em string) para esta molécula.

3 Classes em Python não pagam imposto sobre herança

Objetivo: Modele algum conceito de biologia utilizando herança de classes.

Considerações do experimento: O uso da herança de classes deve fazer sentido dentro do contexto biológico escolhido, isto é, deve haver uma justificativa bem embasada para o uso de herança de classes na sua entrega.

4 `__dunder__`

Objetivo: Se informe sobre métodos dunder que não foram utilizados no material de aula e crie uma classe que use estes métodos.

Considerações do experimento: Use pelo menos 3 métodos dunder que não foram vistos no material da disciplina. Sua classe deve fazer sentido, isto é, não crie uma classe "sem pé nem cabeça" apenas para a entrega. Reflita sobre uma classe onde os métodos dunder propostos realmente fazem sentido. Na sua entrega, explique brevemente o que fazem os métodos dunder que escolheu.

5 Derrube pra fora

Objetivo: implemente o regularizador dropout na rede neural feita em Python puro.

Comentário: algumas tarefas vão apresentar palavras e conceitos que ainda não vimos em sala. Parte do desafio é justamente se informar sobre estes conceitos.

6 Forma, função e ativação

Objetivo: implemente 3 novas funções de ativação na rede neural feita em Python puro. Escreva brevemente sobre elas e a diferenças com relação a função de ativação sigmoidal.

Comentário: aqui não é o lugar de *inventar* funções de ativação. Busque por funções de ativação já existentes utilizadas em redes neurais.

7 Quem classifica a classe classificadora?

Objetivo: altere a rede neural feita em Python puro para resolver um problema de classificação. Treine a rede em um dataset simples de classificação para mostrar que fun-

ciona.

Comentário: aqui é necessário se informar sobre as diferenças de uma rede neural classificadora com relação a uma rede neural regressora. A função de perda, por exemplo, não poderá ser mais a função de perda dos resíduos quadrados.

8 Stop right now, thank you very much

Objetivo: implemente uma estratégia de Parada Antecipada (*Early Stopping*) no processo de treino da rede neural feita em Python puro ou no processo de treino da rede neural feita em PyTorch

Comentário: esta não é para ser resolvida com PyTorch lightning.

9 No alto daquele monte vivia Carlos, o incerto

Objetivo: implemente e compute a incerteza de previsão de uma rede neural utilizando a estratégia de Monte Carlo Dropout.

Sugestão: faça a tarefa "Derrube pra fora" antes de fazer esta.

10 Um momento, por favor!

Objetivo: implemente o otimizador de Descida do Gradiente com Momento (*Gradient Descent with Momentum*) na rede neural feita em Python puro.

11 Tentando uma outra estratégia

Objetivo: crie e treine uma rede neural tipo MLP utilizando o módulo `keras`.