

Relatório 15 - Prática: Redes Neurais Convolucionais 1 (Deep Learning) (II)

Lucas Augusto Nunes de Barros

Descrição das atividades

Aprendizagem supervisionada	Aprendizagem não supervisionada
Redes Neurais Artificiais classificação e regressão	Mapas auto organizáveis detecção de características e agrupamento
Redes Neurais Convolucionais visão computacional	Boltzmann machines sistemas de recomendação redução de dimensionalidade
Redes Neurais Recorrentes análise de séries temporais	Autoencoders redução de dimensionalidade
	Redes adversariais generativas geração de imagens

Os pixels são os menores elementos que compõem uma imagem digital, representando pontos individuais com valores de cor em imagens RGB, onde cada pixel possui 3 valores relativos as cores do sistema (*Red, Green, Blue*), por isso é uma prática comum, converter imagens coloridas em escala de cinza, onde cada pixel tem apenas 1 valor de intensidade. No contexto de *deep learning*, os pixels servem como a entrada para os modelos. Redes neurais convolucionais (CNNs), por exemplo, processam esses pixels para extrair características como bordas, texturas e até padrões mais complexos, que são essenciais para reconhecimento de objetos e tarefas de classificação.

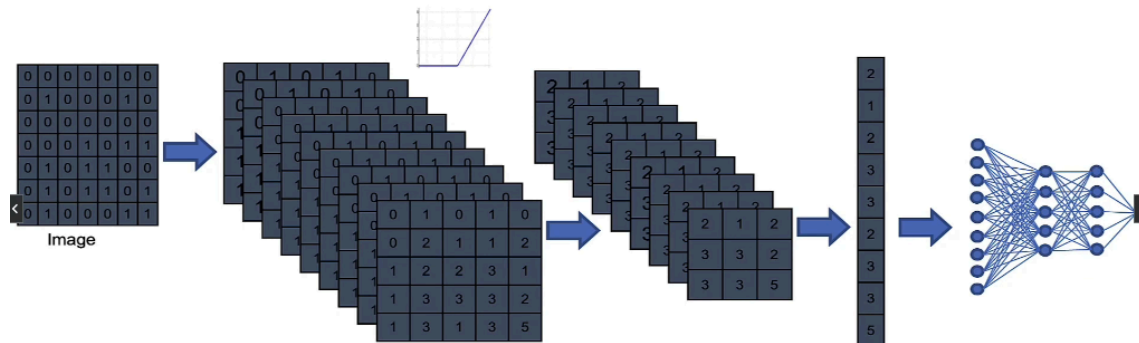
A convolução é uma operação matemática fundamental em *deep learning* que consiste em deslizar um filtro (ou kernel) sobre uma imagem ou outro tipo de dado estruturado, realizando operações elementares entre os valores do filtro e os valores correspondentes da informação analisada. Esse processo é repetido para todas as posições do filtro sobre a imagem, extraindo padrões locais, como bordas, texturas ou formas, dependendo dos valores do filtro. A convolução permite capturar informações espaciais de maneira eficiente, preservando a relação entre pixels vizinhos, além de

diminuir as dimensões da informação analisada. O resultado da convolução, utilizada no contexto de *deep learning*, é conhecido como mapa de características.

A camada de convolução é um dos blocos essenciais em uma CNN, responsável por aplicar o operador de convolução. Ela é composta por múltiplos filtros, cada um aprendendo a detectar diferentes características da entrada. Durante o treinamento, os filtros são ajustados automaticamente para extrair padrões relevantes, como bordas em diferentes orientações ou texturas específicas. A saída da camada de convolução é um conjunto de mapas de características, que representam a presença das características detectadas em diferentes regiões da imagem.

O *pooling* é uma operação usada em redes neurais convolucionais (CNNs) para reduzir a dimensionalidade dos mapas de características, mantendo as informações mais importantes. Ele é aplicado após uma camada de convolução e funciona dividindo o mapa de características em regiões menores resumindo o conteúdo de cada região.

O *flattening* é uma operação comum em redes neurais convolucionais (CNNs) que transforma um mapa de características multidimensionais em um vetor unidimensional. Após as camadas de convolução e *pooling*, os dados ainda estão organizados em matrizes tridimensionais (altura, largura e profundidade). O *flattening* pega esses dados e os "achata" em uma única linha de valores, conectando todos os elementos em sequência. Esse vetor unidimensional é então passado para as camadas densas do modelo. Essa operação essencialmente prepara os dados da camada de convolução (matrizes) para as camadas densas, que operam em formatos vetoriais.



Etapas de processamento de uma CNN

Após os dados serem transformados em um vetor unidimensional, eles são enviados para as camadas densas, que são comuns camadas que consistem em neurônios que se conectam a todos os elementos da próxima camada, permitindo que o modelo combine as características extraídas durante a convolução e o *pooling* de maneira mais complexa.

Nas camadas totalmente conectadas, os dados passam por operações matemáticas e aplicação de funções de ativação. Após esse processamento os dados são enviados para a camada de saída do modelo, onde são produzidos os resultados finais, como por exemplo, as probabilidades de uma imagem possuir objetos X ou Y.

Por vezes, nem sempre mais dados estão disponíveis ou facilmente acessíveis. Como o volume de dados para o treinamento de redes neurais é

consideravelmente grande, é comum o uso de técnicas para aumentar a variabilidade do próprio conjunto de dados existente, no uso de redes convolucionais, uma técnica comum é o *data augmentation*, que consiste aplicar transformações aleatórias, porém realistas, aos dados, como rotações, translações, mudanças de escala, espelhamentos ou ajustes de brilho e contraste em imagens. Essas transformações ajudam a evitar o *overfitting*. Além disso, a técnica torna o modelo mais robusto, permitindo uma melhor generalização para dados não vistos.

Após as aulas teóricas foram implementadas redes neurais de classificação multiclasse e binária. No primeiro exemplo, utilizando a base de dados MNIST, a rede neural foi treinada para classificar algarismos escritos à mão. No Segundo exemplo a rede neural foi treinada para aprender a diferenciar cães e gatos nas imagens.

Conclusão

O curso trouxe de forma prática toda a teoria abordada sobre redes neurais, bem como exemplos práticos e bem detalhados onde cada seção das aulas teóricas era implementada de maneira didática, de forma que facilitou muito o entendimento de conceitos que ainda estavam cinzentos. Mesmo com os exemplos clássicos de modelos de redes neurais é possível perceber o potencial a ser explorado nas mais diversas áreas.

Referências

[1] card15 - [Deep Learning com Python de A a Z - O Curso Completo](#).