

Relatório <11> — <Redes Neurais Convolucionais 1 (Deep Learning) (II)>

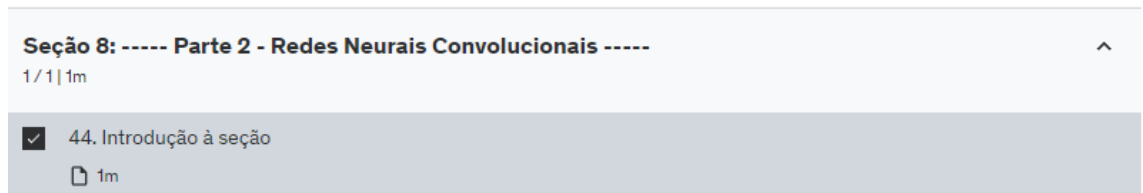
<Jonas Correia>

Descrição da atividade

<Nesta seção o participante deverá descrever com suas próprias palavras os temas abordados nas tarefas do card. É importante demonstrar que houve entendimento dos temas, isto mediante evidências com imagens, gráficos e textos originais.>

Sessão 8

Nesta sessão só é evidenciado o que será apresentado nas sessões seguintes como mostrar a imagem abaixo:



Basicamente será apresentado Redes Neurais Convolucionais:

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

Sessão 9

1. Imagens e Pixels

Nesta aula, foi abordado o conceito fundamental de pixels e sua importância na formação de imagens, especialmente no contexto de visão computacional. Cada imagem é composta por pixels, sendo os menores elementos visíveis, contendo informações sobre cores organizadas em três canais: R (Red), G (Green) e B (Blue). Esses canais representam as cores primárias que, ao serem combinadas em diferentes intensidades, geram todas as outras cores.

Foi explicado que os valores dos canais RGB variam de 0 a 255. Por exemplo, um pixel com os valores (255, 255, 255) é branco, enquanto (0, 0, 0) é preto. Além disso, cada pixel em uma imagem RGB armazena três valores, e o total de informações em uma imagem é obtido multiplicando a largura pela altura e pelo número de canais. No exemplo da aula, uma imagem de 32×32 pixels possui 1024 pixels, resultando em 3072 informações armazenadas no formato RGB.

Também foi demonstrado como funciona a conversão de uma imagem de RGB para escala de cinza. Nesta conversão, os três valores de cada pixel são substituídos por um único valor correspondente à sua intensidade luminosa, reduzindo significativamente o volume de dados armazenados. Por exemplo, uma imagem de 32×32 em escala de cinza armazena apenas 1024 valores, otimizando o processamento computacional. Essa redução de dados é especialmente útil em aplicações como detecção de faces e redes neurais, onde a eficiência no processamento é crucial.

Por fim, foi introduzido o uso de ferramentas para análise de cores, como o “Medidor de Cor Digital”, que ajuda a identificar os valores RGB de um pixel em diferentes partes de uma imagem. Esse tipo de análise é fundamental para compreender a estrutura das imagens e realizar transformações necessárias para algoritmos de visão computacional. A aula encerrou destacando a relevância de entender esses conceitos básicos antes de avançar para tópicos mais complexos, como o classificador Cascade para detecção de faces e objetos.

2. Introdução a redes neurais convolucionais

Nesta aula, foi introduzido o conceito de redes neurais convencionais e sua aplicação em problemas de classificação de imagens. A abordagem começou com a explicação de como imagens, representadas por pixels, podem ser processadas por redes neurais, considerando dois métodos principais:

1. Entrada direta dos pixels:

Aqui, cada pixel da imagem é enviado diretamente para a camada de entrada da rede neural. Por exemplo, em uma imagem 5×5 (25 pixels), cada

pixel, com seu valor correspondente, se torna uma entrada da rede. Essa abordagem é simples, mas não escalável, especialmente para imagens maiores, como as de 32x32, que exigem 1024 valores no formato em escala de cinza ou 3072 no formato RGB.

2. Extração de características:

Nesse método, as características mais relevantes da imagem são extraídas antes de serem enviadas para a rede neural. Um exemplo ilustrado foi a análise das cores específicas associadas a personagens, como Homer e Bart, para identificar suas características distintas. Apesar de útil, essa técnica requer algoritmos manuais de extração de características, tornando-a suscetível a falhas em situações onde as imagens variam ou contêm ruídos.

A evolução desse processo ocorre quando utilizamos redes neurais convencionais, que automatizam a extração de características diretamente das imagens, sem necessidade de intervenção manual. Essas redes aplicam pré-processamentos para identificar padrões únicos em diferentes regiões da imagem, descartando informações redundantes, como bordas ou áreas uniformes. Por exemplo, no caso de emojis, a rede neural focaria em detalhes como olhos, bocas ou lágrimas, ignorando bordas repetitivas.

Os métodos de pré-processamento em redes neurais convencionais incluem operações como convolução, pooling e flattening, que são seguidas por uma rede neural densa tradicional. Essas etapas permitem que o modelo identifique e aprenda automaticamente as características mais relevantes, otimizando o desempenho na classificação de imagens.

A aula destacou ainda como essa abordagem reduz significativamente a necessidade de entradas redundantes, tornando os modelos mais eficientes. Por fim, foi apresentada uma introdução às etapas que compõem o pré-processamento, prometendo explorá-las em detalhe nas próximas aulas, como o operador de convolução, pooling e a camada densa.

3. Etapa 1 - operador de convolução

Nesta aula, aprofundamos o entendimento sobre redes neurais convolucionais (CNNs), focando no operador de convolução. Esse operador é responsável por extrair características relevantes de imagens, reduzindo a quantidade de dados processados enquanto preserva informações cruciais para a classificação. Aqui, detalhamos o funcionamento e a lógica por trás dessa etapa essencial.

Inicialmente, uma imagem é representada como uma matriz de pixels. Por exemplo, uma imagem 7x7 contém 49 pixels. Para iniciar o processo de convolução, aplicamos um **detector de características** (também chamado de kernel ou filtro). Esse detector é uma pequena matriz com valores específicos que

é usada para "varrer" a imagem, executando uma multiplicação elemento por elemento entre as duas matrizes e somando os resultados. O resultado final para cada região analisada é armazenado em uma nova matriz chamada **mapa de características**.

Ao mover o detector de características sobre a imagem (um pixel por vez, ou mais, dependendo da configuração), o processo é repetido para cada região, gerando uma matriz reduzida que retém as características mais importantes da imagem original. Por exemplo, uma matriz 7×7 pode ser reduzida a uma matriz 5×5 , dependendo do tamanho do kernel utilizado e do deslocamento (stride).

Os detectores de características podem ser ajustados para diferentes funções, como:

- **Detecção de bordas**: identifica áreas de contraste na imagem, marcando as bordas.
- **Nitidez (Sharpen)**: Realça detalhes da imagem.
- **Desfoque (Blur)**: Suaviza a imagem, reduzindo detalhes e ruídos.
- **Filtros personalizados**: usados para destacar características específicas, como texturas.

Durante o treinamento da rede neural, os valores dos detectores de características são ajustados automaticamente para otimizar a extração de informações úteis para a tarefa em questão. Assim, cada mapa de características gerado corresponde a uma interpretação única da imagem, extraindo elementos que ajudam a diferenciar as classes.

Após a convolução, aplica-se uma **função de ativação**, geralmente a função ReLU (Rectified Linear Unit). A ReLU transforma os valores negativos do mapa de características em zero, mantendo os valores positivos. Isso ajuda a reduzir a complexidade computacional e torna os dados mais manejáveis, preservando as informações relevantes.

Por fim, a camada de convolução gera múltiplos mapas de características, cada um capturando aspectos diferentes da imagem. Esse conjunto é então utilizado nas etapas subsequentes da rede neural para aprofundar a análise e refinar as previsões. Essa transparência do processo, em que a rede aprende autonomamente os melhores detectores de características, é um dos pontos fortes das redes convolucionais.

O operador de convolução é, portanto, o primeiro passo no processamento de imagens em redes convolucionais, e seu papel é crucial para transformar a imagem original em representações que a rede pode analisar de forma eficiente. Nos próximos passos, exploraremos as demais etapas, como pooling e flattening, para entender o funcionamento completo dessas redes.

4. Etapa 2 - pooling

Nesta aula, exploramos o segundo passo das redes neurais convolucionais (CNNs): o pooling. Essa operação complementa o operador de convolução ao reduzir ainda mais a dimensionalidade dos dados e aumentar a robustez da rede em relação a variações na imagem, como mudanças de escala, posição e orientação.

O pooling permite que a rede reconheça padrões em imagens mesmo que o objeto principal esteja em diferentes contextos ou ângulos. Por exemplo, um cachorro em um quintal, na neve ou em uma casinha deve ser identificado como o mesmo objeto independente do cenário. Essa flexibilidade é fundamental para o sucesso das CNNs em tarefas como classificação de imagens.

A operação de pooling ocorre no mapa de características gerado pela convolução. O tipo mais comum é o max pooling, que trabalha da seguinte forma:

1. É definido o tamanho do pooling, por exemplo, uma matriz de 2×2 .
2. A matriz do pooling é deslocada ao longo do mapa de características em "saltos" (strides), analisando uma região de cada vez.
3. Para cada região, é selecionado o maior valor (max pooling). Esse valor representa as características mais salientes daquela área.
4. O resultado é uma nova matriz reduzida que mantém apenas os valores mais relevantes.

Exemplo prático:

Considere um mapa de características com 25 pixels (5×5). Após aplicar o max pooling com uma matriz 2×2 , o mapa é reduzido para 9 pixels (3×3). Esse processo não apenas diminui a dimensionalidade, mas também realça as características mais marcantes, tornando a rede mais eficiente ao focar nos aspectos essenciais para a diferenciação entre classes.

O pooling é crucial por dois motivos principais:

- Redução de dimensionalidade: diminui o número de parâmetros e cálculos, tornando o treinamento mais rápido e eficiente.
- Invariância espacial: A rede aprende a ignorar pequenas variações no posicionamento, tamanho ou orientação do objeto na imagem, aumentando sua capacidade de generalização.

Embora o max pooling seja o mais utilizado, existem outras variações, como o average pooling, que calcula a média dos valores na região selecionada, suavizando o mapa de características em vez de destacar os picos.

Ao final desta etapa, temos um conjunto de mapas de características reduzidos e otimizados, preparados para avançar para as próximas camadas da rede. Embora neste exemplo utilizemos imagens pequenas, em aplicações reais com imagens de alta resolução, esse processo é escalável e fundamental para lidar com a grande quantidade de dados.

5. Etapa 3 - flattening

A terceira etapa das redes neurais convencionais envolve um processo fundamental para a conexão entre as camadas convolucionais e a rede neural densa. Depois de realizar as etapas anteriores, que consistem nas operações de convolução e pooling, a transformação da matriz resultante em um vetor unidimensional é essencial para dar continuidade ao processamento dos dados.

Inicialmente, a imagem original passa pelo operador de convolução, no qual são aplicados detectores de características específicos. Estes realizam multiplicações ponto a ponto com a imagem, gerando os mapas de características, que destacam os padrões relevantes da imagem. Posteriormente, a função de ativação ReLU é aplicada, eliminando valores negativos e preservando apenas as informações relevantes com valores positivos. Essa camada de convolução forma a base para a extração de características. Em seguida, o pooling é empregado para reduzir ainda mais a dimensionalidade da imagem e enfatizar os aspectos mais importantes, tornando o sistema mais robusto a variações como escala, posição ou orientação da imagem.

O próximo passo após essas operações é converter o resultado obtido, que se encontra no formato de uma matriz bidimensional (o mapa de características reduzido pelo pooling), em um vetor unidimensional. Essa transformação é crucial porque as redes neurais densas, que compõem a próxima camada, recebem como entrada vetores e não matrizes. Durante essa transformação, os valores da matriz são dispostos linearmente em uma sequência ordenada.

Por exemplo, se a matriz reduzida após o pooling tiver valores como 2, 1, 0, 2, 3, 3, 2, 3, 5, essa sequência será reorganizada como um vetor [2, 1, 0, 2, 3, 3, 2, 3, 5]. Esse vetor será usado como entrada para a rede neural densa. O objetivo desse processo é transferir apenas os pixels mais significativos, que encapsulam as características essenciais da imagem, ignorando informações redundantes ou menos relevantes.

Na rede neural densa, os valores do vetor de entrada são processados por uma ou mais camadas ocultas, que podem aplicar funções de ativação, como ReLU, para introduzir não-linearidades e modelar padrões complexos nos dados. Finalmente, na camada de saída, o número de neurônios depende do tipo de problema. Em tarefas de classificação binária, é comum usar um único neurônio na saída com a função de ativação sigmoid, que mapeia os resultados para

valores entre 0 e 1, representando a probabilidade de cada classe. Para problemas de classificação multiclasse, como o reconhecimento de dígitos manuscritos, são usados vários neurônios na camada de saída, um para cada classe, combinados com a função Softmax, que transforma os valores de saída em probabilidades normalizadas.

Esse processo completo ilustra como as redes neurais convencionais convertem uma imagem complexa em representações simplificadas e relevantes, capazes de alimentar modelos densos e permitir previsões precisas.

6. Etapa 4 - rede neural densa

O último passo de uma rede neural convencional é a transição da representação extraída nas camadas convolucionais e de pooling para a camada densa, onde ocorre a classificação final. Este processo culmina com a aplicação de funções de ativação e a interpretação das saídas para produzir probabilidades associadas a cada classe.

Começa-se com a imagem original, que passa pelas etapas iniciais da rede: convolução, ativação e pooling. Essas etapas reduzem a dimensionalidade dos dados, destacam características relevantes e descartam informações redundantes ou menos importantes. O resultado do pooling é uma matriz, que, através do processo de flattening, é transformada em um vetor unidimensional. Este vetor, contendo os valores essenciais da imagem processada, é enviado para a entrada da rede neural densa.

A rede neural densa consiste em camadas completamente conectadas, onde cada neurônio de uma camada está ligado a todos os neurônios da camada seguinte. A primeira camada densa recebe os valores do vetor de entrada, realiza as multiplicações pelos pesos associados, soma os resultados com os valores dos bias e aplica uma função de ativação, como a ReLU, para introduzir não-linearidades. Esse processo é repetido nas camadas ocultas, permitindo à rede modelar padrões mais complexos. A última camada da rede é a de saída, que possui um número de neurônios igual ao número de classes no problema. Cada neurônio na camada de saída corresponde a uma classe específica.

Por exemplo, em um problema de classificação de dígitos escritos à mão (com números como 1, 3 e 9), a camada de saída teria três neurônios. Os valores produzidos por esses neurônios representam a força da associação da entrada processada com cada classe. No entanto, esses valores brutos não são diretamente interpretáveis como probabilidades. Para isso, aplica-se a função de ativação Softmax na camada de saída.

A Softmax transforma os valores brutos dos neurônios em probabilidades normalizadas que somam 1. Isso é feito exponenciando cada valor e dividindo

pelo somatório das exponenciais de todos os valores. Assim, cada saída do Softmax pode ser interpretada como a probabilidade da entrada pertencer a uma classe específica. Se os neurônios associados a uma classe específica apresentam valores significativamente mais altos do que os outros, a probabilidade atribuída a essa classe será maior.

Durante o treinamento, a rede neural ajusta os pesos e os bias de maneira iterativa utilizando o algoritmo de descida do gradiente, com base na minimização de uma função de perda, como a entropia cruzada, que mede a discrepância entre as previsões da rede e as classes reais. Além disso, os detectores de características aplicados nas camadas convolucionais também são ajustados. A rede experimenta diferentes detectores de características e retém aqueles que maximizam seu desempenho na tarefa, aprendendo automaticamente os padrões mais eficazes para distinguir entre as classes.

Por exemplo, a rede pode aprender que, para identificar o dígito "1", certos neurônios devem apresentar valores altos, enquanto os valores de outros permanecem baixos. Da mesma forma, para o dígito "3", um conjunto diferente de neurônios será ativado com valores altos. Este padrão é identificado e refinado durante o treinamento.

No final do processo, a rede é capaz de classificar novas imagens atribuindo probabilidades a cada classe com base nos padrões aprendidos. Esta capacidade de generalização reflete o objetivo principal das redes neurais convolucionais: extrair, comprimir e interpretar informações relevantes de dados de entrada complexos para fornecer previsões precisas.

7. Teste 2: Teoria sobre redes neurais convolucionais



Bom trabalho!

Pergunta 1:

Considere as afirmações abaixo:

- I. Redes neurais convolucionais podem ser utilizadas para problemas da área de visão computacional
- II. Uma rede neural convolucional não requer a utilização de uma rede neural densa (fully connected)
- III. Supondo que temos uma imagem com a dimensão 5 x 5 (25 pixels no total), quando utilizamos uma rede neural convolucional todos os 25 pixels são utilizados na camada de entrada da rede neural densa
- IV. Uma rede neural convolucional procura identificar quais são as características mais importantes das imagens utilizando detectores de características
- V. As etapas de uma rede neural convolucional são respectivamente: operador de convolução, pooling, flattening e rede neural densa



Apenas as afirmações I, IV e V estão corretas



Apenas as afirmações I, II e V estão corretas



Apenas as afirmações III, IV e V estão corretas



Todas as afirmativas estão corretas



Bom trabalho!

Pergunta 2:

Considere as afirmações abaixo:

- I. O operador de convolução reduz a dimensionalidade da imagem com o intuito de capturar as características mais importantes
- II. Os detectores de características mais relevantes para tarefas de classificação de imagens são aqueles que realçam as bordas dos objetos
- III. A camada de convolução de uma rede neural convolucional é composta por vários mapas de características
- IV. A utilização da função max pooling é preferível do que usar funções de média ou mínimo porque ela realça as características mais importantes dos objetos
- V. O treinamento em uma rede neural convolucional é realizado pelo algoritmo de descida do gradiente e além de atualizar os pesos da rede neural densa, é também necessário atualizar o mapa de características

☐ Somente as afirmações I, II e III estão corretas

☐ Somente as afirmações I, IV e V estão corretas

☐ Somente as afirmações III, IV e V estão corretas

☒ Todas as afirmações estão corretas



Bom trabalho!

Pergunta 3:

Considerando a representação abaixo de uma imagem e um detector de características, indique os valores dos mapas de características resultante da multiplicação das duas matrizes.

1	0	0	0	0	0	0
0	1	0	1	0	1	0
0	0	0	0	0	0	0
0	0	1	1	1	1	1
0	1	1	1	1	0	0
0	1	0	1	1	0	1
0	1	0	0	0	1	1

Imagem

X

1	0	0
0	1	1
0	1	0

=

Detector de características
(feature detector)

☐ 2 1 0 1 1
0 2 3 2 1
2 3 3 5 2
0 2 4 3 2
2 1 3 2 0

☒ 2 1 1 1 1
0 2 1 2 1
2 3 3 3 2
3 2 4 3 1
2 2 3 2 3



Bom trabalho!

Pergunta 4:

Qual a matriz resultante da aplicação da operação de max pooling no mapa de características abaixo?

2	1	1	1	1
0	2	1	2	1
2	3	3	3	2
3	2	4	3	1
2	2	3	2	3

Mapa de características
(feature map)



2 2 1
3 4 2
2 3 3



Bom trabalho!

Pergunta 5:

Considerando o mapa de características abaixo já com a aplicação da operação de max pooling, indique o resultado do vetor com a aplicação do operador de flattening

2	2	1
3	3	2
2	3	3

☐ 1 2 2 2 3 3 3 2

☐ 1 2 3 2 3 3 2 3 2

☒ 2 2 1 3 3 2 2 3 3

☐ 2 3 2 2 3 3 1 2 3

Sessão 10

A sessão 10 tem como foco o desenvolvimento e aprofundamento dos conceitos previamente abordados ao longo do curso ou do material de estudo. Neste contexto, foi dada ênfase à aplicação prática dos conhecimentos adquiridos, por meio da implementação de códigos que demonstram de forma concreta os conceitos teóricos explorados.

Códigos desenvolvidos: [git/car15](https://github.com/car15)

Conclusões

Ao longo destas sessões, exploramos os conceitos fundamentais das Redes Neurais Convolucionais (CNNs), suas etapas de processamento, e aplicações práticas. As CNNs demonstraram ser ferramentas poderosas no campo da visão computacional, permitindo a extração automática de características relevantes das imagens por meio de operações como convolução, pooling e flattening.

Esses métodos destacam a eficiência e a robustez das CNNs em lidar com grandes volumes de dados, reduzindo a dimensionalidade e aumentando a capacidade de generalização. A integração com redes densas na etapa final complementa o modelo, garantindo alta precisão em tarefas de classificação.

A compreensão e a implementação prática desses conceitos oferecem uma base sólida para resolver problemas complexos, como reconhecimento de padrões e classificação de imagens. Este estudo reforça a relevância das CNNs em aplicações do mundo real, desde a detecção de objetos até a análise avançada de imagens em diversos domínios.