

Resolução de Problemas com Deep Learning

VISÃO COMPUTACIONAL E REDES CONVOLUCIONAIS (CNNs)

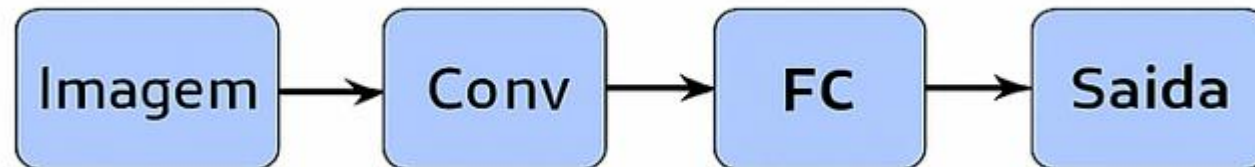
Prof. Bruno Martins, Msc.

Estrutura das CNNs

As CNNs possuem uma arquitetura modular. Cada parte da rede tem um papel específico:

- Camadas Convolucionais: aplicam filtros que detectam padrões visuais locais
- Pooling: reduz a resolução da imagem mantendo os padrões mais fortes
- Fully Connected: transformam tudo em vetores para realizar a classificação final

🧠 Analogia: Como se fosse uma equipe: uns observam pistas, outros resumem e o chefe decide o que está na imagem.



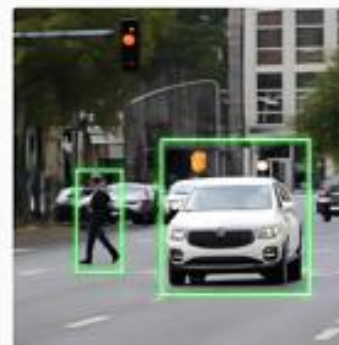
Aplicações em Visão Computacional



As CNNs estão por trás de muitas tecnologias:

- Diagnóstico por imagem (raio-x, tomografia)
- Reconhecimento facial (celular, redes sociais)
- Carros autônomos (detecção de faixa, semáforos)
- E-commerce visual (reconhecimento de produto)
- Câmeras inteligentes (segurança e vigilância)

💡 Reflexão: Onde mais você já viu IA analisando imagens?



Transfer Learning

Treinar uma CNN do zero é caro e demorado. Com Transfer Learning, usamos redes já treinadas:

- As primeiras camadas ficam congeladas (detectam bordas, formas)
- Ajustamos apenas as últimas camadas para a nova tarefa
- Acelera o processo e melhora a performance com poucos dados

⚙️ Exemplo: Usar uma CNN treinada em ImageNet para classificar imagens de folhas.

Transfer Learning e Modelos Pré-Treinados



Modelos Famosos de CNNs

Modelos e características principais:

- VGG16: Arquitetura sequencial e profunda
- ResNet: Skip connections, redes muito profundas
- EfficientNet: Equilíbrio entre profundidade e custo

- VGG: ideal para aprendizado
- ResNet: robusta para profundidade extrema
- EfficientNet: muito eficiente para produção

Modelo	Profundidade	Diferencial
VGG16	Alta	Estrutura sequencial
ResNet	Muito Alta	Skip connections
EfficientNet	Balanceada	Alta eficiência com menos custo

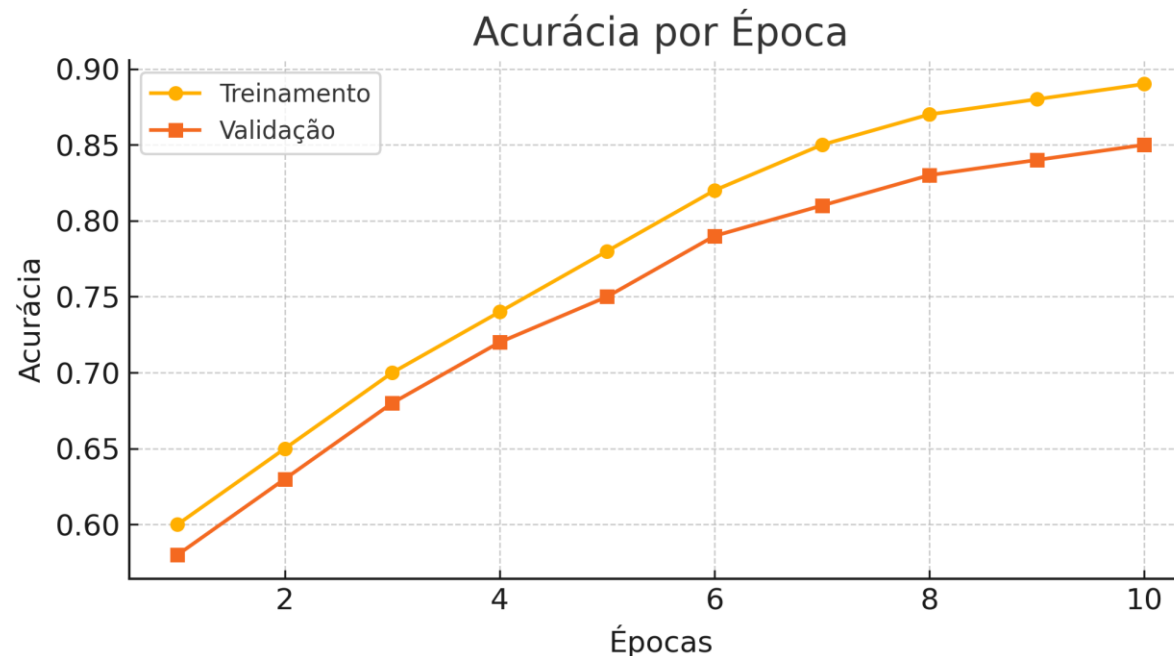


Atividade Prática – Implementação de uma CNN

Objetivo: Implementar, treinar e avaliar uma CNN com MNIST ou CIFAR-10.

Etapas:

1. Exploração do dataset: visualizar e entender as classes
2. Construção da rede: montar camadas Conv, Pooling, Dense
3. Treinamento: configurar otimizador e métricas, treinar por épocas
4. Avaliação: analisar gráficos de acurácia e perda
5. Predição: testar com novas imagens e comparar rótulos



✓ Importante: Observar sinais de overfitting e discutir soluções como dropout e data augment.



Dopout

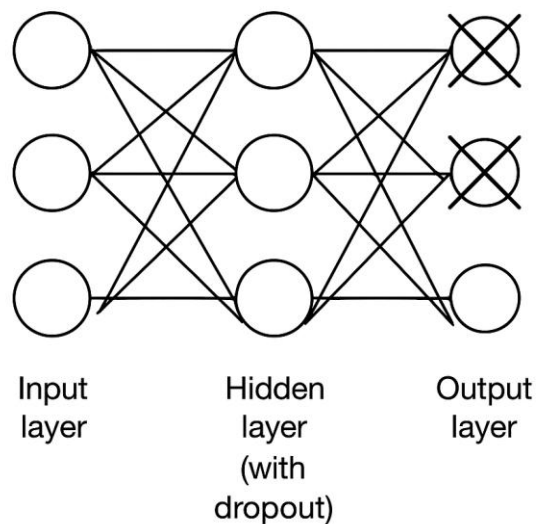
Dropout é uma **técnica de regularização** usada durante o treinamento de redes neurais para **evitar overfitting**.

💡 Como funciona?

Durante o treinamento, **alguns neurônios são desativados aleatoriamente** em cada iteração. Isso força a rede a **não depender demais de alguns caminhos específicos**, tornando o modelo mais robusto.

💡 **Analogia:** é como se você treinasse um time de futebol fazendo revezamento entre os jogadores. Cada treino tem uma formação diferente, o que obriga o time inteiro a se adaptar e melhorar como um grupo.

📊 Exemplo visual:



Data Augmentation


O que é?

Data Augmentation é uma técnica para **aumentar artificialmente a quantidade de dados de treino** gerando **variações** de cada imagem original.

Como funciona?

Aplica-se **transformações simples** nas imagens, como:

- Rotação
- Corte (crop)
- Zoom
- Inversão horizontal (flip)
- Alteração de brilho ou contraste

 Exemplo: uma imagem de um gato virada, cortada ou com brilho alterado continua sendo um gato, e isso ajuda o modelo a **generalizar melhor**.

Por que usar?

- Aumenta o número de exemplos únicos
- Reduz o risco de overfitting
- Torna o modelo mais robusto a variações reais (como posição, iluminação, etc.)



O que é o CIFAR-10?

CIFAR-10 (Canadian Institute for Advanced Research) é um **conjunto de dados amplamente utilizado em tarefas de classificação de imagens**, especialmente em redes neurais convolucionais (CNNs).











Ele foi criado por pesquisadores da Universidade de Toronto e é frequentemente usado para **testar e comparar o desempenho de modelos de visão computacional**.

Características do CIFAR-10

Atributo	Valor
 Tipo de dado	Imagens coloridas (RGB)
 Tamanho da imagem	32 x 32 pixels
 Número de canais	3 (vermelho, verde, azul)
 Número de classes	10
 Uso principal	Classificação de imagens
 Total de imagens	60.000 (50.000 treino + 10.000 teste)

As 10 classes do CIFAR-10

Cada imagem é rotulada como pertencente a **uma dessas 10 categorias**:

-  airplane (avião)
-  automobile (carro)
-  bird (pássaro)
-  cat (gato)
-  deer (veado)
-  dog (cachorro)
-  frog (sapo)
-  horse (cavalo)
-  ship (navio)
-  truck (caminhão)



💡 Por que é tão usado?

- **Pequeno e leve:** fácil de treinar, até mesmo em máquinas simples.
- **Desafiador o suficiente:** apesar do tamanho reduzido, os objetos estão em diferentes posições, cores e ângulos.
- **Ideal para CNNs:** ótimo para aprender arquitetura e experimentar ajustes.