Documentación: Redes Neuronales Convolucionales (CNN)

Introducción

Una red neuronal convolucional (CNN, por sus siglas en inglés Convolutional Neural Network) es una arquitectura especializada de red neuronal profunda diseñada para procesar datos que tienen una estructura de tipo grid, como una imagen. Las CNNs son ampliamente utilizadas en tareas de visión por computadora, como clasificación de imágenes, detección de objetos, segmentación semántica, entre otras.

Características clave

1. **Extracción automática de características**: A diferencia de los modelos tradicionales que requieren ingeniería manual de características, las CNNs aprenden automáticamente representaciones jerárquicas de los datos.
2. **Parámetros compartidos**: Los mismos pesos del filtro (kernel) se aplican en múltiples posiciones de la entrada, reduciendo drásticamente el número de parámetros.
3. **Conectividad local**: Cada neurona en una capa convolucional está conectada solamente a una región local de la entrada, lo que permite capturar patrones espaciales locales.
4. **Profundidad**: Las CNNs modernas tienen múltiples capas, lo que permite aprender representaciones de bajo nivel (como bordes) hasta conceptos de alto nivel (como formas o clases).

Estructura de una CNN

Una CNN típica consta de una secuencia de capas específicas que transforman los datos de entrada en una representación de salida útil para tareas como la clasificación.

1. Capa de Entrada

- Recibe una matriz de dimensiones (altura × anchura × canales), por ejemplo, una imagen de tamaño 64×64 con 3 canales RGB.
- No realiza procesamiento; simplemente representa el dato inicial.

2. Capas Convolucionales (Convolutional Layers)

- Aplican filtros (también llamados kernels) que se deslizan sobre la entrada.
- Cada filtro extrae una característica específica (bordes, texturas, etc.).
- El resultado de aplicar un filtro es un mapa de activación (feature map).
- Fórmula típica para el número de parámetros de un filtro:
  (tamaño del filtro) × (número de canales de entrada) + 1 (bias)

3. Función de Activación (comúnmente ReLU)

- Se aplica a la salida de la convolución para introducir no linealidad.
- ReLU (Rectified Linear Unit) transforma los valores negativos en cero:
  f(x) = max(0, x)

4. Capas de Pooling (submuestreo)

- Reducen la dimensionalidad espacial de los mapas de activación.
- Ayudan a controlar el sobreajuste y reducen el coste computacional.
- Tipos comunes:
  - Max Pooling: selecciona el valor máximo en una región.
  - Average Pooling: calcula el promedio de la región.

5. Capas Completamente Conectadas (Fully Connected, FC)

- Conectan todas las neuronas de la capa anterior con todas las de la siguiente.
- Funcionan como una red neuronal tradicional y se utilizan al final del modelo.
- Son responsables de producir la predicción final (por ejemplo, la clase de una imagen).

6. Capa de Salida

- Generalmente una capa densa con activación softmax para clasificación multiclase.
- La función softmax transforma las salidas en probabilidades:
  softmax(z_i) = exp(z_i) / Σ exp(z_j) para todos los valores z en la salida.

Proceso de Entrenamiento

1. **Propagación hacia adelante (forward pass)**: los datos se procesan capa por capa hasta obtener una predicción.
2. **Cálculo de la función de pérdida**: se mide el error entre la predicción y la etiqueta verdadera.
3. **Retropropagación (backpropagation)**: se calcula el gradiente del error respecto a cada parámetro usando la regla de la cadena.
4. **Actualización de parámetros**: se ajustan los pesos usando un optimizador como SGD o Adam.

Ventajas

- Eficiencia en procesamiento de datos espaciales como imágenes.
- Menor número de parámetros comparado con redes densas tradicionales.
- Excelentes resultados en tareas de visión por computadora.

Desventajas

- Requieren grandes cantidades de datos y poder de cómputo.
- Son menos interpretables que los modelos tradicionales.
- Sensibles a cambios en la orientación o posición del objeto a menos que se use data augmentation.

Aplicaciones comunes

- Clasificación de imágenes (por ejemplo, reconocimiento de dígitos, rostros, etc.).
- Detección de objetos (por ejemplo, autos en una calle).
- Reconocimiento de texto manuscrito.
- Segmentación semántica y médica.
- Sistemas de recomendación visual.