Нейронные сети для изображений

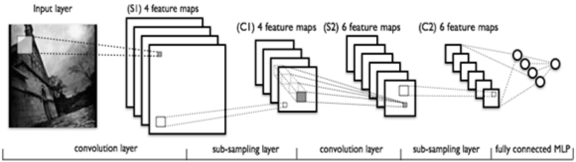
Лунев Иван,
Петраков Михаил

Санкт-Петербург
2019г.

Структура CNN

Сверточная нейронная сеть состоит из разных видов слоев:

- сверточные (convolutional) слои
- объединяющие слои (pooling layer)
- полносвязные слои



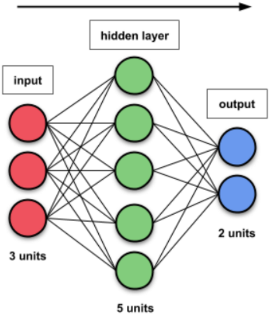
Количество параметров

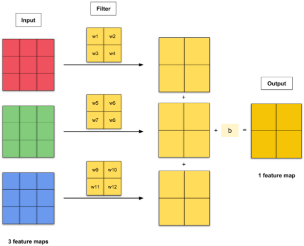
В снс количество параметров сокращается.

Рассчитаем кол-во параметров для нс:

$n = (i * h + h * o) + (h + o)$, где i – размер входного слоя, h – размер скрытого слоя, o – размер выходного слоя

$n = (3 * 5 + 5 * 2) + (5 + 2) = 32$





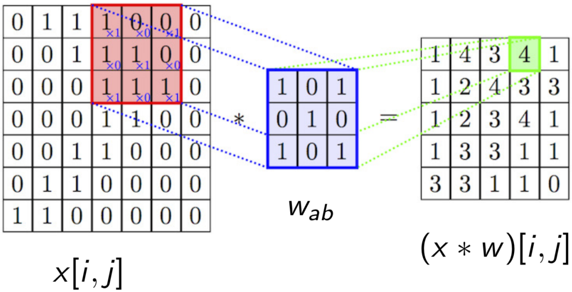
Рассчитаем кол-во параметров для снс:

$$n = [x * (w * w) * o] + o = [3 * (2 * 2) * 1] + 1 = 13$$

Сверточный слой

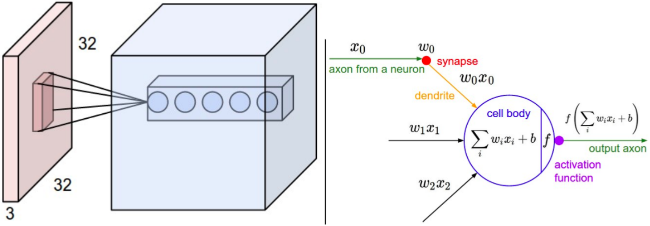
$x[i, j]$ - исходные признаки, пиксели $n \times m$ изображения;
w_{ab} - ядро свертки, где считаем, что ядро является прямоугольным, а a и b длины его сторон;
Сверточный нейрон:

$$(x * w)[i, j] = \sum_a \sum_b w_{ab} x[i + a, j + b]$$



Сверточный слой (Пример)

Например, пусть вход имеет размер $[32 \times 32 \times 3]$. В этом случае возьмем размер фильтра $[5 \times 5 \times 3]$, то есть фильтр будет иметь форму прямоугольного параллелепипеда, в общей сложности $5 * 5 * 3 = 75$ весовых коэффициентов (и $+1$ параметр смещения). Используются параллельно несколько разных фильтров, за счет чего сеть растет "вглубь".



Сверточный слой (Пример)

Гиперпараметры, формирующие размер выхода сверточного слоя:

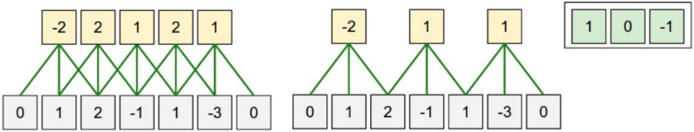
- глубина (depth) — количество разных фильтров
- шаг (stride) — шаг сдвига фильтра
- дополнение нулями (zero-padding)

Для одномерного случая: $(W - F + 2P)/S + 1$ - размер выходного слоя, где W - размер входа, F - ширина фильтра, P — заполнение 0, S — шаг.

$F = 3$, $W = 5$, $P = 1$.

left - $S = 1$, следовательно размер выхода $(5 - 3 + 2)/1 + 1 = 5$

right - $S = 2$, следовательно размер выхода $(5 - 3 + 2)/2 + 1 = 3$



Сверточный слой (Итоги)

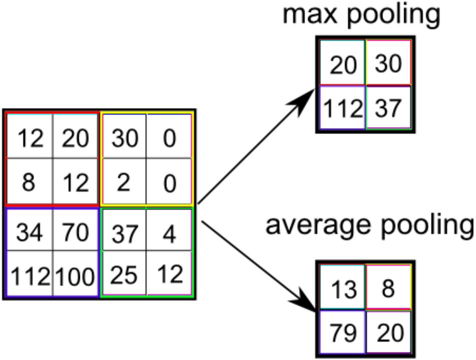
- вход размером $W_1 \times H_1 \times D_1$ (обычно рассматриваются картинки, у которых глубина (RGB) равна $D_1 = 3$)
- требует 4 гиперпараметра :
  - K – количество фильтров
  - F – размер фильтра (имеется ввиду трехмерный квадратный фильтр со стороной F и глубиной $D_1 = 3$, но вообще говоря форма может быть любой)
  - S – шаг свертки
  - P – заполнение нулями (ширина полосы по кругу картинки, которые заполняются нулями)
- на выходе $W_2 \times H_2 \times D_2$
  - $W_2 = (W_1 - F + 2P)/S + 1$
  - $H_2 = (H_1 - F + 2P)/S + 1$
  - $D_2 = K$
- $F * F * D_1$ весов на фильтр, всего $F * F * D_1 * K$ весов
- Далее к получившимся элементам сверточного слоя применяют функцию активации. Обычно берут Выпрямитель: $\text{ReLu}(p) = \max(0, p)$;

Объединяющий слой (pooling layer)

Объединяющий слой нейронов – это необучаемая свёртка с шагом $h > 1$, агрегирующая данные прямоугольной области $h \times h$:

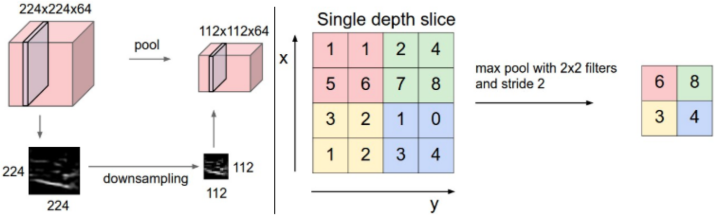
$$y[i,j] = F(x[hi,hj],\ldots,x[hi+h-1,hj+h-1]),$$

гду F – агрегирующая функция: max, average и т.п.



Объединяющий слой (pooling layer)

- вход $W_1 \times H_1 \times D_1$
- 2 гиперпараметра:
  - F – ширина квадратного фильтра
  - S – шаг фильтра
- Выход $W_2 \times H_2 \times D_2$:
  - $W_2 = (W_1 - F)/S + 1$
  - $H_2 = (H_1 - F)/S + 1$
  - $D_2 = D_1$
- не принято дополнять входной объект нулями
- чаще всего $F = 3, S = 2$ или $S = 2, F = 2$



Полносвязный слой

Последний из типов слоев это слой обычного многослойного персептрона. Цель слоя – классификация, моделирует сложную нелинейную функцию, оптимизируя которую, улучшается качество распознавания. Вычисление значений нейрона можно описать формулой:

$$\mathrm{x}_\mathrm{j}^\mathrm{l} = \sigma(\sum_\mathrm{i} \mathrm{x}_\mathrm{i}^{\mathrm{l}-1} * \mathrm{w}_{\mathrm{i,j}}^{\mathrm{l}-1} + \mathrm{b}_\mathrm{j}^{\mathrm{l}-1}),$$

где

- x_j^l – карта признаков j (выход слоя l),
- $\sigma()$ – функция активации,
- b^l – коэффициент сдвига слоя l,
- $\mathrm{w}_{\mathrm{i,j}}^\mathrm{l}$ – матрица весовых коэффициентов слоя l.

Выделяются следующие функции активации:

- сигмойда: $\sigma(z) = \frac{1}{1-e^{-az}}$, $a \in \mathbb{R}$;
- гиперболический тангенс: $\sigma(z) = \frac{e^{az}-e^{-az}}{e^{az}+e^{-az}}$;
- softmax: $\sigma(z)_{\mathsf{i}} = \frac{e^{z_{\mathsf{i}}}}{\sum\limits_{k=1}^{K} e^{z_k}}$.

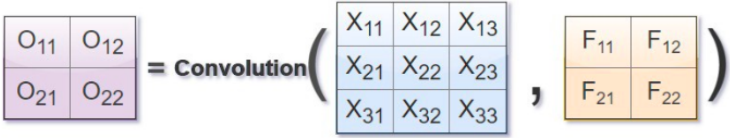
По сути, к снс применим обычный алгоритм прямого и обратного распространения, так как она подходит по определение обычной нс, мы просто обнуляем большинство весов, которые скорее всего не дают вклада.

Но за счет специфичного построения последующих слоев, у нас появляется возможность переписать эти алгоритмы в более удобную для вычислений форму.

Далее представлен алгоритм, по которому усовершенствуются Forwardpropagation и Backpropagation.

Forwardpropagation

Операция свертки может быть записана так, как описано на рисунке ниже.



$$O_{11} = F_{11}X_{11} + F_{12}X_{12} + F_{21}X_{21} + F_{22}X_{22}$$

$$O_{12} = F_{11}X_{12} + F_{12}X_{13} + F_{21}X_{22} + F_{22}X_{23}$$

$$O_{21} = F_{11}X_{21} + F_{12}X_{22} + F_{21}X_{31} + F_{22}X_{32}$$

$$O_{22} = F_{11}X_{22} + F_{12}X_{23} + F_{21}X_{32} + F_{22}X_{33}$$

Forwardpropagation

Теперь, чтобы вычислить градиенты фильтра F относительно ошибки E, необходимо решить уравнения, которые можно записать в форме операции свертки.

$$\begin{bmatrix} \partial E/\partial F_{11} & \partial E/\partial F_{12} \\ \partial E/\partial F_{21} & \partial E/\partial F_{22} \end{bmatrix} = \textbf{Convolution}\left(\begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \\ X_{31} & X_{32} & X_{33} \end{bmatrix}, \begin{bmatrix} \partial E/\partial O_{11} & \partial E/\partial O_{12} \\ \partial E/\partial O_{21} & \partial E/\partial O_{22} \end{bmatrix} \right)$$

Точно так же мы можем найти градиенты входной матрицы X относительно ошибки E.

$$\frac{\partial E}{\partial X_{11}} = \frac{\partial E}{\partial O_{11}} F_{11} + \frac{\partial E}{\partial O_{12}} 0 + \frac{\partial E}{\partial O_{21}} 0 + \frac{\partial E}{\partial O_{22}} 0$$

$$\frac{\partial E}{\partial X_{12}} = \frac{\partial E}{\partial O_{11}} F_{12} + \frac{\partial E}{\partial O_{12}} F_{11} + \frac{\partial E}{\partial O_{21}} 0 + \frac{\partial E}{\partial O_{22}} 0$$

$$\frac{\partial E}{\partial X_{13}} = \frac{\partial E}{\partial O_{11}} 0 + \frac{\partial E}{\partial O_{12}} F_{12} + \frac{\partial E}{\partial O_{21}} 0 + \frac{\partial E}{\partial O_{22}} 0$$

$$\frac{\partial E}{\partial X_{21}} = \frac{\partial E}{\partial O_{11}} F_{21} + \frac{\partial E}{\partial O_{12}} 0 + \frac{\partial E}{\partial O_{21}} F_{11} + \frac{\partial E}{\partial O_{22}} 0$$

$$\frac{\partial E}{\partial X_{22}} = \frac{\partial E}{\partial O_{11}} F_{22} + \frac{\partial E}{\partial O_{12}} F_{21} + \frac{\partial E}{\partial O_{21}} f_{12} + \frac{\partial E}{\partial O_{22}} F_{11}$$

$$\frac{\partial E}{\partial X_{23}} = \frac{\partial E}{\partial O_{11}} 0 + \frac{\partial E}{\partial O_{12}} F_{22} + \frac{\partial E}{\partial O_{21}} 0 + \frac{\partial E}{\partial O_{22}} F_{11}$$

$$\frac{\partial E}{\partial X_{31}} = \frac{\partial E}{\partial O_{11}} 0 + \frac{\partial E}{\partial O_{12}} 0 + \frac{\partial E}{\partial O_{21}} F_{21} + \frac{\partial E}{\partial O_{22}} 0$$

$$\frac{\partial E}{\partial X_{32}} = \frac{\partial E}{\partial O_{11}} 0 + \frac{\partial E}{\partial O_{12}} 0 + \frac{\partial E}{\partial O_{21}} F_{22} + \frac{\partial E}{\partial O_{22}} F_{21}$$

$$\frac{\partial E}{\partial X_{33}} = \frac{\partial E}{\partial O_{11}} 0 + \frac{\partial E}{\partial O_{12}} 0 + \frac{\partial E}{\partial O_{21}} 0 + \frac{\partial E}{\partial O_{22}} F_{22}$$

Backpropagation

Теперь вышеприведенные вычисления могут быть получены с помощью операции свертки другого типа, известной как полная свертка. Чтобы получить градиенты входной матрицы, необходимо повернуть фильтр на 180 градусов и рассчитать полную свертку повернутого фильтра по градиентам выходного сигнала относительно ошибки.

Полная свертка может быть визуализирована как выполнение процедуры, представленной на рисунке ниже.