

# Topics in Social Data Science

w.

University of Copenhagen

Complementary course notes

Thor Donsby Noe \*

February 27, 2019

---

\*Department of Economics, University of Copenhagen, Øster Farimagsgade 5, DK-1353 Copenhagen K, Denmark (e-mail: [jwz766@alumni.ku.dk](mailto:jwz766@alumni.ku.dk))

"..."

---

# 1 CONVOLUTIONAL- & RECURRENT NEURAL NETWORKS

## 1.1 Convolutional Nets

Issue with vanilla feed-forward neural network

- Data squeezed into a  $1 \times N$  vector.
- Single operation on whole input.
- No attention to **spatial adjacency**.

Solution: 'Slide' weight matrix over a part of the input. E.g.

- Input image:  $32 \times 32 \times 3$  (i.e. RGB-colors)
- Filter:  $5 \times 5 \times 3 \rightarrow \text{dot product} \rightarrow 1$  number
- Activation map:  $28 \times 28 \times 1$  (convolved over all spatial locations)
  - One activation map for each filter. These are stacked.
  - E.g.  $28 \times 28 \times 6$  for 6  $5 \times 5 \times 3$  filters.
    - \* An additional activation map can be added by using a filter on the activation map of the former layer.
    - \* E.g. a new activation map layer:  $24 \times 24 \times 10$  by using 10  $5 \times 5 \times 3$  filters.

### Stride

- How many pixels are moved each time, e.g. 1.

## Padding

- The no. zeroes added at the borders of the input image before applying the filter.
- Full padding: If you don't want the image to shrink.
- Half padding: The image shrinks less, e.g. padding = 1.

Width of the activation map will be

$$W = \frac{N + 2P - F}{S} + 1$$

E.g. for the example above with padding=2

$$W = \frac{32 + 2 \cdot 2 - 5}{1} + 1 = 32$$

**Pooling:** Condense the information by downsampling.

- Speeds up the learning process.
- 'max pool' where only largest value is saved for each quadrant as they are expected to contain the more important signal.
- 'average pooling'
- E.g.  $16 \times 16 \rightarrow 4 \times 4$
- Doesn't shrink it in the depth direction.

## 1.2 Recurrent Nets

Optimal for **sequential data** but useful for non-sequential data as well.

Issue for **all** feed forward neural networks

- Input and output must be of hard-assigned dimensions.

Backpropagation: Compute the sum of losses.

- Can get complicated as the number becomes very high up with long chains.
- Can break it into parts.

- f: Forget vector (0 = forget, 1 = remember) → what to forget from c-vector

– c: New hidden vector, i.e. long term memory

– c is not converted, → not multiplied by weight → gradient doesn't explode → much quicker

- i, g: adds to memory vector, c

- o: output vector