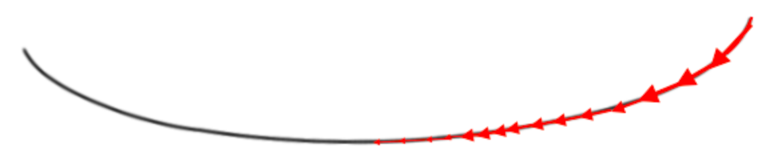
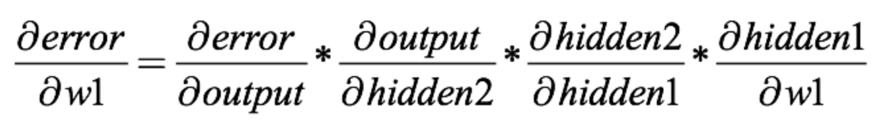
NN Basic

Jo MinKi

1. Drop out
2. Overlapping pooling
3. Local Response Normalization (Local contrast normalization)
4. At least inside of this Deep learning field, Kernel = Feature = Window = Filter   
   <http://deeplearning.stanford.edu/wiki/index.php/File:Convolution_schematic.gif>   
   The Yello one is that.
5. Basic structure for CNN   
   Feature extraction layer: Convolution layer and pooling layer  
   Classifier layer: Fully connected layer
6. Deep NN   
   When the depth goes deeper and deeper, it gets easier to prone overfitting so it requires sensitive and appropriate modification and initialization. And it needs a huge amount of computation so you can’t just make it deeper and deeper. (But the recent model ResNet has over 150 layer)
7. Gradient vanishing and ReLU  
     
   As you see when the point approach to minimum than the gradient gets smaller. And if you try back propagation with sigmoid or tanh, the total gradient is multiplications of smaller numbers than 1/4. (Because the range of derivative of sigmoid is [0,1/4].) So the size of the gradient is too much small, it is never going to improve.   
   But we can solve this problem with ReLU. The derivative of ReLU is always 1 or 0, so it is never vanishs if the input is positive. But one caution of this is that when the input gets negative number, the gradient die. It always gives same result ‘0’ when it takes negative number, and you cannot recover this again. So if you care about this issue, you should keep your input values to positive or normalize them.   
   <https://ayearofai.com/rohan-4-the-vanishing-gradient-problem-ec68f76ffb9b>