Midterm

Notebook:	ANLY 590 w/ Joshuah		
Created:	10/24/2018 3:29 PM	**Updated:**	10/25/2018 1:25 PM
Author:	1253836247@qq.com		
URL:	https://georgetown.instructure.com/courses/60318/discussion_topics/210481		

1. Limitation of NNet

The graph above can be composed in multiple layers to create hierarchical networks, i.e. networks with more than one layer. Intermediate layers are hidden in the sense that they are never observed. This poses a statistical problem because it means that we will never be able to identify the true parameters of the network (because they can be interchanged among nodes).

Note: Identifiability is important in statistics since it ensures with a large enough sample we'll be able to identify the true values of our model. That is we'll be able to know with certainty things like the mean and standard deviation. Since neural networks have non-identifiable parameters we'll never be able to learn their true values. In practice this doesn't pose impede the inferential process since we usually seek approximates for the true value. In particular we trade the ability of our models to generalize well for exacctness. This is usually sufficient since finding the optimal parameter set doesn't yield significant gains over local optima.

2. Activation function:
    1. Sigmoid:
        - Drawback:
            - Sigmoids saturate for large values of jxj which and "kill" gradients:
                - When saturated no information is propageted across nodes because the gradient is effectively
                - When intializing wieghts caution must be taken not to saturate the neuron
                - If the intial wieghts are too large then the network will not update
            - Sigmoid outputs are not zero-centered:
                - During training this tends to bias the dynamics of graident descent (i.e. neurons always positive or negative)
                - Unstable updates for the gradient
                - Less severe than gradient saturiation
    2. Hyperbolic Tangent (Tanh).
        - Non-linear, Maps real-valued numbers to the range [-1, 1]
        - tanh neuron is a scaled sigmoid neuron, in particular the following holds: tanh(x) =2*sigmoid*(2x)^(-1)
        - Its activations can saturate and output is 0 centered
        - Tanh non-linearity is always preferred to the sigmoid nonlinearity
    3. ReLU
        - Linear unit: f(x) = max(0, x)
        - Popular in the last few years, Pros:
            - Linear so non-saturating
            - Greatly accelerates the convergence of stochastic gradient descent compared to sigmoid/tanh functions.
            - Inexpensive operations to calculate the threshold matrix for gradients when compared to tanh/sigmoid neurons

            - ReLu ReLU can be implemented by thresholding a matrix of activations at zero.
            - Sigmoid/TanH involve expensive operations such as exponentiation
    - Cons:
        - ReLU units can be fragile during training and "die"
        - For large gradients, weight updates may cause ReLU neurons to never activate
        - Subsequent gradient updates will be 0 and the neuron will effectively be dead
        - This can occur in 40% of your network and is known as the 'dying ReLu' problem
        - Setting the learning rate thoughtfully is therefore important

4. Leaky ReLus:
    - Piecewise liner
    - Variant of ReLu to fix the "dying ReLU" problem
    - Includes a parameter *a* that tunes the slope of the unit for values less than 0
    - $f(x) = 1(x < 0)(ax) + 1(x >= 0)(x)$
5. Maxout:
    1. $f(x) = \max(w^{T}_{1} * x_1 + b_1, w^{T}_{2} * x2 + b_2)$
6. Note:

- It is rare to mix different neurons in the same network (although nothing saying you can't)
- "What neuron type should I use?" use **ReLU**
- Considerations:
    - Careful with your learning rates
    - Monitor the fraction of "dead" units in a network
        - If dead units or setting the learning rate is difficult consider leaky ReLu or maxout units
        - **Never use sigmoid**
        - If you want to use a sigmoid function use **tanh**
        - Results with tanh will still be worse than ReLU/Maxout.