The following are notes I took as I read the GELU paper.

1 Abstract and Introduction

The GELU activation function is $x\Phi(x)$ where $\Phi(x)$ is the standard Gaussian cumulative distribution function. The GELU nonlinearity weights inputs by their value, rather than gates inputs by their sign as in ReLUs. The paper performs an empirical evaluation of the GELU nonlinearity against the ReLU and ELU activations.

As networks became deeper, sigmoid activations proved less effective and less-probabilistic for training compared to ReLU (makes hard gating decisions based upon an input's sign: $1\ if\ x > 0\ else\ 0$. GELU relates to stochastic regularizers (?) - expectation of a modification to Adaptive Dropout, leading to a more probabilistic view of a neuron's output.

2 GELU Formation

- ReLU and dropout both yield a neuron's output
- ReLU does this deterministically while dropout does this randomly
- GELU multiplies the input by zero or one stochasictaclly by multiplying the neuron input x by $m \sim \text{Bernoulli}(\Phi(x)) = P(X \leq x),\ X \sim \mathcal{N}(0, 1)$.
- as x decreases, inputs have a higher probability of being 'dropped"
- due to wanting a deterministic decision[1] from a neural network we can take the expected transformation of the regularizer on an in put x where we have $\Phi(x) \times 1x + (1 - \Phi(x)) \times 0x = x\Phi(x)$
- We can approximate the GELU with

$$\text{GELU}(x) = xP(X \leq x) = x\Phi(x) = x \cdot \frac{1}{2}\left[1 + \text{erf}\left(\frac{x}{\sqrt{2}}\right)\right]$$

3 GELU Experiments

3.1 MNIST Classification

- Evaluated GELU against ELU and ReLU
- Each 8-layer 128 neuron wide neural network is trained for 50 epochs with a batch size of 128
- uses Adam optimizer instead of SGD w/o momentum
- weights are initialized with unit norm rows, as this has positive impact on each nonlinearity's performance[2]
- Tuned over the learning rates { 10^{-3}, 10^{-4}, 10^{5} } with 5k validation examples from the training set and take the median results for five runs
- GELU tends to have lowest mediam training log logg w/ and w/o dropout (Figure 2)
- GELU is more robust to noised inputs (Figure 3)

3.2 MNIST Autoencoder

- Train a deep autoencoder on MNIST with layers of width 1000, 500, 250, 30, 250, 500, 1000 in that order
- Use Adam optimizer and batch size of 64
- Loss is MSE
- Learning rate is varied from 10^{-3} to 10^{-4}

[1] an input should not have multiple different outputs, also leads to more stability as a slight change in the input should not cause a drastic change in the output

[2] need to dig into this a little more

3.3 CIFAR-10/100

4 Discussion

- GELU performed previous nonlinearities across several experiments
- However as $\sigma \to 0$ and $\mu = 0$ then the GELU becomes a ReLU[3]

[3] how does the GELU become a ReLU?