1 FP of VNN: Column Version

Have L hidden layers
for k = 0

$$\boldsymbol{h}^{(0)}(\boldsymbol{x}) = \boldsymbol{x}$$

for k form 1 to L
a) Hidden layer pre-activation

$$\boldsymbol{a}^{(k)}(\boldsymbol{x}) = W^{(k)}\boldsymbol{h}^{(k-1)}(\boldsymbol{x}) + \boldsymbol{b}^{(k)}$$

b) Hidden layer activation

$$\boldsymbol{h}^{(k)}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{a}^{(k)}(\boldsymbol{x}))$$

for k = L+1
Output layer activation

$$\boldsymbol{a}^{(L+1)}(\boldsymbol{x}) = W^{(L+1)}\boldsymbol{h}^{(L)}(\boldsymbol{x}) + \boldsymbol{b}^{(L+1)}$$

$$\hat{\boldsymbol{y}} = \boldsymbol{h}^{(L+1)}(\boldsymbol{x}) = softmax(\boldsymbol{a}^{L+1}(\boldsymbol{x}))$$

2 BP of VNN: Column Version

$$\nabla_{\boldsymbol{a}^{(L+1)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \hat{\boldsymbol{y}} - \boldsymbol{y}$$

For k from L+1 to 1
a) compute gradient of hidden layer parameter

$$\nabla_{W^{(k)}} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \nabla_{\boldsymbol{a}^{(k)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) * \boldsymbol{h}^{(k-1)}(\boldsymbol{x})^T$$

$$\nabla_{\boldsymbol{b}^{(k)}} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \nabla_{\boldsymbol{a}^{(k)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y})$$

b) compute gradient of hidden layer below(stop when it comes to $\boldsymbol{x} = \boldsymbol{h}^{(0)}(\boldsymbol{x})$)

$$\nabla_{\boldsymbol{h}^{(k-1)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) = (W^{(k)^T}) * (\nabla_{\boldsymbol{a}^{(k)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}))$$

c) compute gradient of hidden layer below(before activation)

$$\nabla_{\boldsymbol{a}^{(k-1)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \nabla_{\boldsymbol{h}^{(k-1)}(\boldsymbol{x})} Loss(\hat{\boldsymbol{y}}, \boldsymbol{y}) \odot [..., f\prime(a^{(k-1)}(\boldsymbol{x})_j), ...]$$

3 FP of VNN: Row Version

Have L hidden layers
for k = 0

$$\boldsymbol{d}^{(0)}(\boldsymbol{v}) = \boldsymbol{v}$$

for k form 1 to L
a) Hidden layer pre-activation

$$\boldsymbol{z}^{(k)}(\boldsymbol{v}) = \boldsymbol{d}^{(k-1)}(\boldsymbol{v})U^{(k)} + \boldsymbol{c}^{(k)}$$

b) Hidden layer activation

$$\boldsymbol{d}^{(k)}(\boldsymbol{v}) = \boldsymbol{g}(\boldsymbol{z}^{(k)}(\boldsymbol{v}))$$

for k = L+1
Output layer activation

$$\boldsymbol{z}^{(L+1)}(\boldsymbol{v}) = \boldsymbol{d}^{(L)}(\boldsymbol{v})U^{(L+1)} + \boldsymbol{c}^{(L+1)}$$

$$\hat{\boldsymbol{o}} = \boldsymbol{d}^{(L+1)}(\boldsymbol{x}) = softmax(\boldsymbol{z}^{L+1}(\boldsymbol{v}))$$

4 BP of VNN: Row Version

$$\nabla_{\boldsymbol{z}^{(L+1)}(\boldsymbol{v})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) = \hat{\boldsymbol{o}} - \boldsymbol{o}$$

For k from L+1 to 1
a) compute gradient of hidden layer parameter

$$\nabla_{U^{(k)}} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) = \boldsymbol{d}^{(k-1)}(\boldsymbol{v})^T * \nabla_{\boldsymbol{z}^{(k)}(\boldsymbol{v})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o})$$

$$\nabla_{\boldsymbol{c}^{(k)}} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) = \nabla_{\boldsymbol{z}^{(k)}(\boldsymbol{v})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o})$$

b) compute gradient of hidden layer below(stop when it comes to $\boldsymbol{v} = \boldsymbol{d}^{(0)}(\boldsymbol{v})$)

$$\nabla_{\boldsymbol{d}^{(k-1)}(\boldsymbol{v})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) = (\nabla_{\boldsymbol{z}^{(k)}(\boldsymbol{x})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o})) * ({U^{(k)}}^T)$$

c) compute gradient of hidden layer below(before activation)

$$\nabla_{\boldsymbol{z}^{(k-1)}(\boldsymbol{v})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) = \nabla_{\boldsymbol{d}^{(k-1)}(\boldsymbol{x})} CE(\hat{\boldsymbol{o}}, \boldsymbol{o}) \odot [..., g\prime(z^{(k-1)}(\boldsymbol{v})_j), ...]$$

5 VNN: Example for L=1

$$\nabla_{\boldsymbol{z}_2} CE = \hat{\boldsymbol{o}} - \boldsymbol{o}$$

$$\nabla_{U_2} CE = \boldsymbol{d}_1^T * \nabla_{\boldsymbol{z}_2} CE$$

$$\nabla_{\boldsymbol{c}_2} CE = \nabla_{\boldsymbol{z}_2} CE$$

$$\nabla_{\boldsymbol{d}_1} CE = \nabla_{\boldsymbol{z}_2} CE * U_2^T$$

$$\nabla_{\boldsymbol{z}_1} CE = \nabla_{\boldsymbol{d}_1} CE \odot \sigma'(\boldsymbol{z}_1)$$

$$\nabla_{U_1} CE = \boldsymbol{v}^T * \nabla_{\boldsymbol{z}_1} CE$$

$$\nabla_{\boldsymbol{c}_1} CE = \nabla_{\boldsymbol{z}_1} CE$$

Notice: $\sigma'(\boldsymbol{z}_1) = sigmoid_grad(\boldsymbol{d}_1)$