4 Deep neural networks

Problem 4.1 Consider composing the two neural networks in figure 4.8. Draw a plot of the relationship between the input x and output y' for $x \in [-1, 1]$.

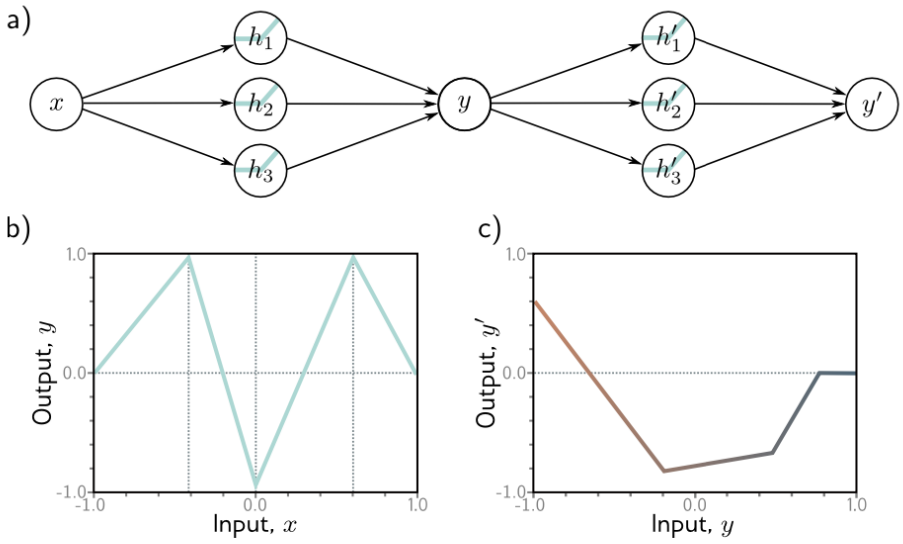
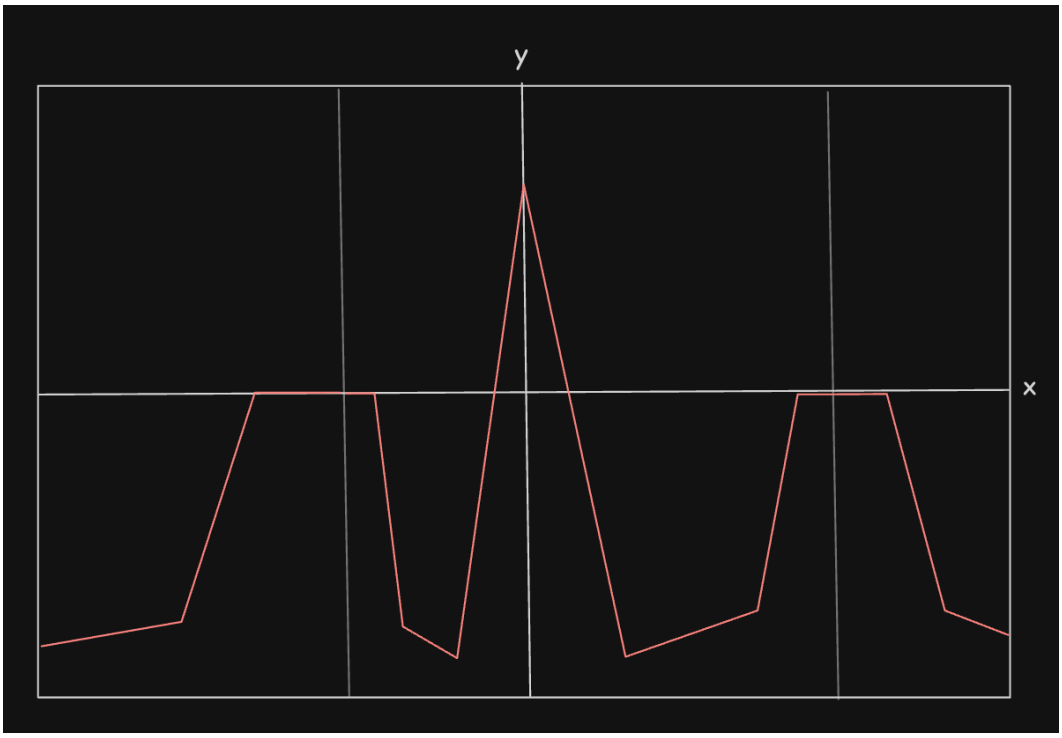


Figure 4.8 Composition of two networks for problem 4.1. a) The output y of the first network becomes the input to the second. b) The first network computes this function with output values $y \in [-1, 1]$. c) The second network computes this function on the input range $y \in [-1, 1]$.



Problem 4.2 Identify the four hyperparameters in figure 4.6.

Number of layers ($K = 3$) and the width of each hidden layer ($D_1 = 4, D_2 = 2, D_3 = 3$).

Problem 4.3 Using the non-negative homogeneity property of the ReLU function (see problem 3.5), show that:

$$\begin{aligned}&\text{ReLU}\left[\boldsymbol{\beta}_1 + \lambda_1 \cdot \boldsymbol{\Omega}_1 \text{ReLU}[\boldsymbol{\beta}_0 + \lambda_0 \cdot \boldsymbol{\Omega_0}\mathbf{x}]\right] = \\ &\lambda_0\lambda_1 \cdot \text{ReLU}\left[\frac{1}{\lambda_0\lambda_1}\boldsymbol{\beta}_1 + \boldsymbol{\Omega}_1 \text{ReLU}[\frac{1}{\lambda_0}\boldsymbol{\beta}_0 + \boldsymbol{\Omega_0}\mathbf{x}]\right]\end{aligned}$$

where λ_0 and λ_1 are non-negative scalars. From this, we see that the weight matrices can be rescaled by any magnitude as long as the biases are also adjusted, and the scale factors can be re-applied at the end of the network.

$$
\begin{aligned}
&\lambda_0\lambda_1 \cdot \text{ReLU}\left[\frac{1}{\lambda_0\lambda_1}\boldsymbol{\beta}_1 + \boldsymbol{\Omega}_1\text{ReLU}[\frac{1}{\lambda_0}\boldsymbol{\beta}_0 + \boldsymbol{\Omega_0}\mathbf{x}]\right] \\
&\quad= \text{ReLU}\left[\lambda_0\lambda_1\frac{1}{\lambda_0\lambda_1}\boldsymbol{\beta}_1 + \lambda_0\lambda_1\boldsymbol{\Omega}_1\text{ReLU}[\frac{1}{\lambda_0}\boldsymbol{\beta}_0 + \boldsymbol{\Omega_0}\mathbf{x}]\right] \\
&\quad= \text{ReLU}\left[\boldsymbol{\beta}_1 + \lambda_1\boldsymbol{\Omega}_1\text{ReLU}[\lambda_0\frac{1}{\lambda_0}\boldsymbol{\beta}_0 + \lambda_0\boldsymbol{\Omega_0}\mathbf{x}]\right] \\
&\quad= \text{ReLU}\left[\boldsymbol{\beta}_1 + \lambda_1\boldsymbol{\Omega}_1\text{ReLU}[\boldsymbol{\beta}_0 + \lambda_0\boldsymbol{\Omega_0}\mathbf{x}]\right]
\end{aligned}
$$

Problem 4.4 Write out the equations for a deep neural network that takes $D_i = 5$ inputs, $D_o = 4$ outputs and has three hidden layers of sizes $D_1 = 20, D_2 = 10$, and $D_3 = 7$, respectively, in the forms of equation 4.15 and 4.16. What are the sizes of the weight matrix $\boldsymbol{\Omega}_\bullet$ and bias vector $\boldsymbol{\beta}_\bullet$?

In form of equation 4.15:

$$
\begin{aligned}
\mathbf{h}_1 &= \mathbf{a}[\boldsymbol{\beta}_0 + \boldsymbol{\Omega}_0\mathbf{x}] \\
\mathbf{h}_2 &= \mathbf{a}[\boldsymbol{\beta}_1 + \boldsymbol{\Omega}_1\mathbf{h}_1] \\
\mathbf{h}_3 &= \mathbf{a}[\boldsymbol{\beta}_2 + \boldsymbol{\Omega}_2\mathbf{h}_2] \\
\mathbf{y} &= \boldsymbol{\beta}_3 + \boldsymbol{\Omega}_3\mathbf{h}_3
\end{aligned}
$$

and in form of equation 4.16:

$$
\mathbf{y} = \boldsymbol{\beta}_3 + \boldsymbol{\Omega}_3\mathbf{a}[\boldsymbol{\beta}_2 + \boldsymbol{\Omega}_2\mathbf{a}[\boldsymbol{\beta}_1 + \boldsymbol{\Omega}_1\mathbf{a}[\boldsymbol{\beta}_0 + \boldsymbol{\Omega}_0\mathbf{x}]]]
$$

where

$$\begin{aligned}
\boldsymbol{\Omega_0} &\in \mathbb{R}^{20\times 5} \\
\boldsymbol{\Omega_1} &\in \mathbb{R}^{10\times 20} \\
\boldsymbol{\Omega_2} &\in \mathbb{R}^{7\times 10} \\
\boldsymbol{\Omega_3} &\in \mathbb{R}^{4\times 7} \\
\boldsymbol{\beta_0} &\in \mathbb{R}^{20} \\
\boldsymbol{\beta_1} &\in \mathbb{R}^{10} \\
\boldsymbol{\beta_2} &\in \mathbb{R}^{7} \\
\boldsymbol{\beta_3} &\in \mathbb{R}^{4}
\end{aligned}$$

Problem 4.5 Consider a deep neural network with $D_i = 5$ inputs, $D_o = 1$ output, and $K = 20$ hidden layers containing $D = 30$ hidden units each. What is the depth of the network? What is the width?

This network has depth of 20 and width of 30.

Problem 4.6 Consider a network with $D_i = 1$ input, $D_o = 1$ output, and $K = 10$ layers, with $D = 10$ hidden units in each. Would the number of weights increase more if we increased the depth by one or the width by one? Provide your reasoning.

If we add a hidden layer, there are 10x10 = 100 weights added from the fully connected layer and 10 more biases, so 110 increase of parameters in total.

If we add a hidden unit to each layer, the number of FC connections between each of the hidden layers increase from 10x10 to 11x11, together with the first and last connection requiring 2 more parameters. So 21x9 + 2 = 191 more weights. Also, we need 10 more biases. So 201 increase of parameters in total.

Problem 4.7 Choose values for the parameters $\boldsymbol{\phi} = \{\phi_0, \phi_1, \phi_2, \phi_3, \theta_{10}, \theta_{11}, \theta_{20}, \theta_{21}, \theta_{30}, \theta_{31}\}$ for the shallow neural network in equation 3.1

(with ReLU activation functions) that will define an identity function over a finite range $x \in [a, b]$.

Let $\phi_2 = \phi_3 = 0$ to consider only the activation from the first hidden unit. Then, equation 3.1 can be written as

$$\mathbf{y} = \phi_0 + \phi_1 \mathbf{a}[\theta_{10} + \theta_{11} x]$$

Note that the ReLU activation function can produce only nonnegative output. To create an identity function over a range of $x \in [a, b]$, we can setup the linear unit feeded to the activation function by $\theta_{10} + \theta_{11} x = -a + x$ so that it stays above 0. Then, we let $\phi_0 = a, \phi_1 = 1$ to map the function back to the identity function.

Problem 4.8 Figure 4.9 shows the activations in the three hidden units of a shallow network (as in figure 3.3). The slopes in the hidden units are 1.0, 1.0, and -1.0, respectively, and the "joints" in the hidden units are at positions 1/6, 2/6, and 4/6. Find values of ϕ_0, ϕ_1, ϕ_2, and ϕ_3 that will combine the hidden unit activations as $\phi_0 + \phi_1 h_1 + \phi_2 h_2 + \phi_3 h_3$ to create a function with four linear regions that oscillate between output values of zero and one. The slope of the leftmost region should be positive, the next one negative, and so on. How many linear regions will we create if we compose this network will itself? How many will we create if we compose it with itself K times?

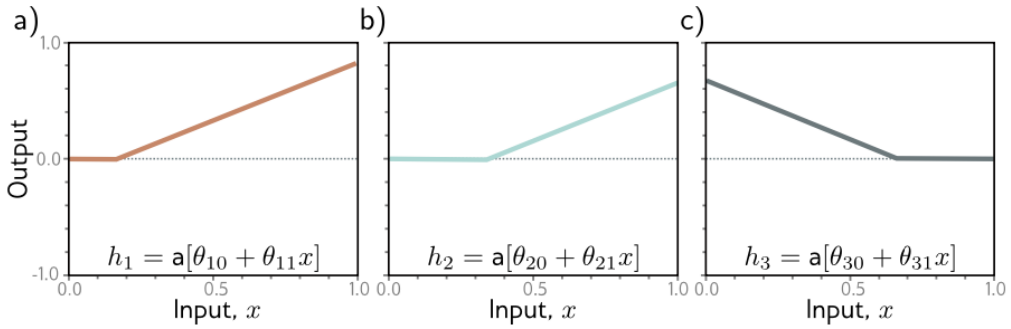
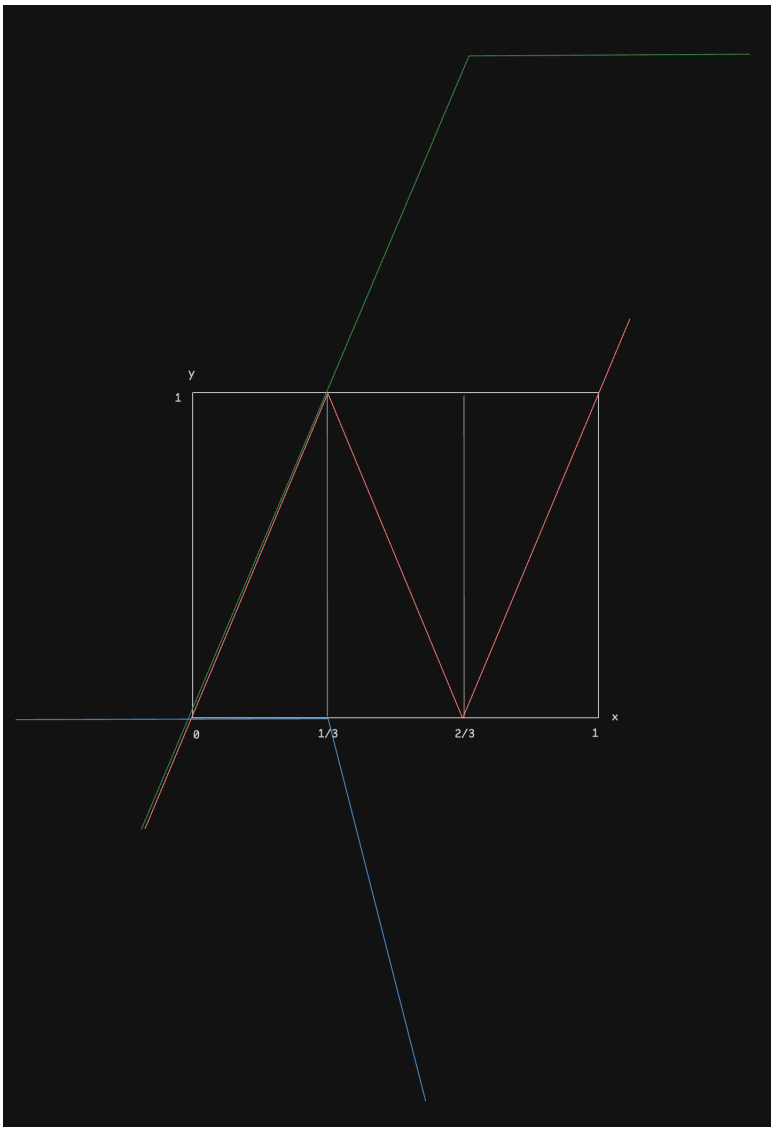


Figure 4.9 Hidden unit activations for problem 4.8. a) First hidden unit has a joint at position $x = 1/6$ and a slope of one in the active region. b) Second hidden unit has a joint at position $x = 2/6$ and a slope of one in the active region. c) Third hidden unit has a joint at position $x = 4/6$ and a slope of minus one in the active region.

$\phi_0 = 4, \phi_1 = -12, \phi_2 = 9, \phi_3 = -6$ values produce the desired output. Composing this function with itself will produce a 4-fold output, so there will be 16 linear regions. Likewise, there will be 4^K linear regions if composed K times.

Problem 4.9 Following problem 4.8, is it possible to create a function with three linear regions that oscillates back and forth between output values of zero and one using a shallow network of two linear units? Is it possible to create a function with five linear regions that oscillates in the same way using a shallow network with four hidden units?



Given only two hidden units, it is forced that the first ReLU activation (the green plot) to solely produce the output in the first linear region while the second hidden unit (the blue plot) is inactive. Since the slope of the second linear unit is also nonzero, the joint between the first and second linear region should be produced by the blue plot. However, in this setup we cannot make the slope of the last linear region to be positive. Therefore, in general we cannot produce the desired output with two hidden units.

However, it is possible to create five oscillating linear regions with four hidden units.

Problem 4.10 Consider a deep neural network with a single input, a single output, and K hidden layers, each of which contains D hidden units. Show that this network will have a total of $3D + 1 + (K-1)D(D+1)$ parameters.

- Between the input and first hidden layer: D slopes, D biases
- Between the output and last hidden layer: D slopes, 1 bias
- Between each hidden layers: D^2 slopes, D biases → $(K-1)D(D+1)$ parameters

Overall, we have $3D + 1 + (K-1)D(D+1)$ parameters.

Problem 4.11 Consider two neural networks that map a scalar input x to scalar output y. The first network is shallow and has $D = 95$ hidden units. The second is deep and has $K = 10$ layers, each containing $D = 5$ hidden units. How many parameters does each network have? How many linear regions can each network make (see equation 4.17)? Which would run faster?

Parameters: using the formula from problem 4.10,

- first network: $3 \cdot 95 + 1 = 286$
- second network: $3 \cdot 5 + 1 + 9 \cdot 5 \cdot 4 = 196$

Linear regions: using equation 4.17,

- first network: 96 linear regions.
- second network:

$$(5+1)^{(10-1)} \cdot \left(\binom{5}{0} + \binom{5}{1} \right) = 6^{10}$$

Also shallow networks would run faster, since the computation can be done in parallel, even though it has more parameters and thus more work to do.