Now we are going to study the paper:
Petersen & Voigtlaender, 2018
"Optimal approximation of piecewise smooth functions using deep ReLU networks"

Idea: Study piecewise constant (piecewise smooth) label functions, e.g.,

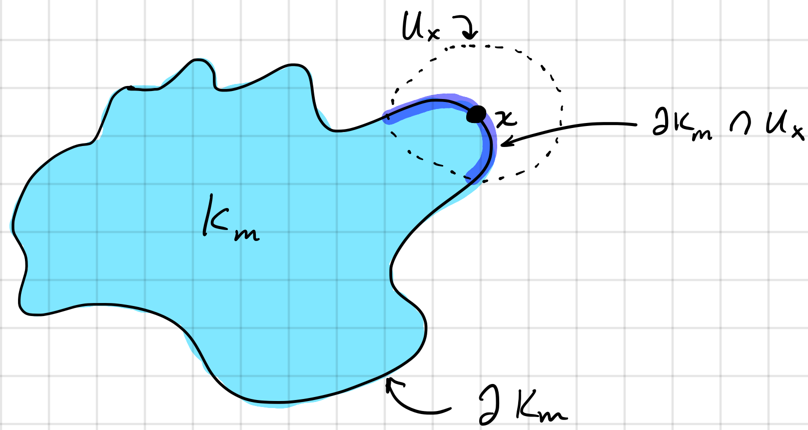
$$F(x) = \sum_{m=1}^{M-1} m \, \mathbb{1}_{K_m}(x), \qquad \mathbb{1}_{K_m}(x) = \begin{cases} 1, & x \in K_m \\ 0, & x \notin K_m \end{cases}$$

which models a classification problem with M classes.
Throughout the data (test) space will be

$$\mathcal{X} = \left[-\tfrac{1}{2}, \tfrac{1}{2}\right]^d \subset \mathbb{R}^d$$

with $K_m \subseteq \mathcal{X}$ and $K_\ell \cap K_m = \phi$ for all $\ell \neq m$

Recall we considered a similar model in Cybenko (1989) as well.
We will assume $\underbrace{\partial K_m}_{\text{boundary of } K_m} \in C^\alpha$ is α-smooth



Note: $\partial K \in C^\alpha$ means the following: For any point $x \in \partial K$, there is an open neighborhood around x, say U_x, for which $\partial K \cap U_x$ is the graph of a function $\phi \in C^\alpha(\mathbb{R}^{d-1})$ on some open subset $V \subset \mathbb{R}^{d-1}$, that is:
$\partial K \cap U_x = \{(z, \phi(z)) : z \in V\}$ (up to a change of coordinates)

Summary of results:

(i) The # of layers depends on d and α (not the # of neurons)
(ii) The # of nonzero weights in the neurons depends on the desired accuracy ε and d and α
(iii) The results are sharp if one limits the magnitude of the weights (recall Pinkus 1999, weights really large)
(iv) If the label function F has a suitable low dimensional structure, the network can take advantage of it (not unlike the results of Poggio, et al. 2017)

Now let us be more precise.

Label function $F : [-\frac{1}{2}, \frac{1}{2}]^d \to \mathbb{R}$ of the form:

$$F(x) = \sum_{m=1}^{M-1} a_m \underbrace{\mathbb{1}_{K_m}(x)}, \qquad a_m \in \mathbb{R}, \quad K_m \subseteq [-\tfrac{1}{2}, \tfrac{1}{2}]^d, \quad \partial K_m \in C^\alpha$$

Can be generalized to smooth functions

Neural network:

$$f(x;\theta) = A_L \circ \sigma \circ A_{L-1} \circ \cdots \circ \sigma \circ A_1(x)$$

$$A_\ell(x) = W_\ell x + b_\ell, \quad W_\ell \in \mathbb{R}^{d_\ell \times d_{\ell-1}}, \quad b_\ell \in \mathbb{R}^{d_\ell}, \quad d_0 = d$$

$$\sigma(z) = \max(0, z) = \mathrm{ReLU}(z)$$

# of neurons $= \sum_{\ell=1}^{L} d_\ell$

# of nonzero weights $= \|\theta\|_0 = \sum_{\ell=1}^{L} \left(\|W_\ell\|_0 + \|b_\ell\|_0 \right)$

where $\|z\|_0 = \#\{ z(i) \neq 0 \}$

Furthermore, the weights are **bounded and quantized.** This means they can be stored on a computer. More precisely, the neural network $f(x;\theta)$ has (t,ε)-quantized weights if all entries of A_ℓ and b_ℓ, for $1 \leq \ell \leq L$, are elements of

$$\underbrace{[-\varepsilon^{-t}, \varepsilon^{-t}]}_{\text{bounded}} \cap \underbrace{2^{-t\lceil \log_2(1/\varepsilon) \rceil}\mathbb{Z}}_{\text{quantized}}$$

Note: $c \cdot \mathbb{Z} = \{ cn : n \in \mathbb{Z} \}$

Error measured in root mean squared loss:

$$\|F - f\|_2 = \|F - f\|_{L^2([-\frac{1}{2},\frac{1}{2}]^d)} = \left[\int_{[-\frac{1}{2},\frac{1}{2}]^d} |F(x) - f(x;\theta)|^2 \, dx \right]^{1/2}$$

Can be generalized to $L^p([-\frac{1}{2},\frac{1}{2}]^d)$, $p \in (0, \infty)$

There are lots of new things to consider. Let's first look at a functional class similar to one we have seen before, and see the effect of quantization.

Define: $C^\alpha(\mathcal{X})$, for $\alpha = s + \gamma$, $s \in \mathbb{N}$, $0 < \gamma \leq 1$, as the space of functions $F : \mathcal{X} \to \mathbb{R}$ for which

$$\|F\|_\alpha = \|F\|_{C^\alpha(\mathcal{X})} = \max\left\{ \max_{\substack{\beta \in \mathbb{N}^d : \\ \|\beta\|_1 \leq s}} \|\partial^\beta F\|_\infty \, , \, \max_{\substack{\beta \in \mathbb{N}^d : \\ \|\beta\|_1 = s}} \mathrm{Lip}_\gamma(\partial^\beta F) \right\} < +\infty$$

where

$$\mathrm{Lip}_\gamma(G) = \sup_{\substack{x, x' \in \mathcal{X} \\ x \neq x'}} \frac{|G(x) - G(x')|}{\|x - x'\|_2^\gamma}$$

Theorem: Let $F \in C^\alpha([-\frac{1}{2}, \frac{1}{2}]^d)$ with $\|F\|_\alpha = B < \infty$ and let $0 < \varepsilon < 1/2$. Then there are constants $t = t(d, \alpha, B) \in \mathbb{N}$ and $c = c(d, \alpha, B) > 0$ for which there is a neural network $f(x; \theta)$ with

$$L \leq (2 + \lceil \log_2 \alpha \rceil)(11 + \alpha/d)$$

layers and

$$\|\theta\|_0 \leq c \cdot \varepsilon^{-d/\alpha}$$

nonzero, (t, ε)-quantized weights such that

$$\|F - f\|_{L^2} < \varepsilon$$

and

$$\|f\|_{L^\infty} \leq \lceil B \rceil$$

Remarks:

(a) # of nonzero weights similar to previous complexity results

(b) Smoother F, i.e., larger α, more layers needed, but less total # of nonzero weights

(c) Weights are bounded and quantized, and $\sigma = \mathrm{ReLU}$, so the 2-layer result of Pinkus does not apply.

Idea of proof:

(a) ReLU NNs can approximately implement multiplication. In particular, let $z, z' \in [-\frac{1}{2}, \frac{1}{2}]$. Then there is a ReLU NN, $g(z, z'; \theta)$, with L layers and

$$\|\theta\|_0 = O(\varepsilon^{-c/L}), \quad c > 0 \text{ universal},$$

such that

$$|zz' - g(z, z'; \theta)| \leq \varepsilon \quad \text{for all } z, z' \in [-\tfrac{1}{2}, \tfrac{1}{2}]$$

The weights θ are also (t, ε)-quantized

(b) Now one can approximate monomials

(c) Now one can approximate polynomials, including Taylor polynomials

↓

(d) Patch together several local Taylor polynomial approximations of F to get an approximation of F on all of $[-\frac{1}{2}, \frac{1}{2}]^d$. To carry out this step, one has to show neural networks can implement cutoff functions using a fixed number of layers and weights. That is, ReLU neural networks with $c = c(d)$ nonzero, (s, ε)-quantized weights, $s = s(d)$, can approximate functions of the form

$$G(x) = \mathbb{1}_{[a_1, b_1] \times \cdots \times [a_d, b_d]}(x)$$

to ε accuracy in the $L^2[-\frac{1}{2}, \frac{1}{2}]^d$ norm.

Now let us go back to label functions of the form:

$$F(x) = \sum_{m=1}^{M} a_m \mathbb{1}_{K_m}, \quad \partial K_m \in C^\alpha \qquad (*)$$

We will build up to a result for functions F of the form $(*)$.

Step 1 : Horizon functions : A horizon function is 0-1 valued function with a jump along a hypersurface such that the jump surface is the graph of a smooth fcn.

Define the heavyside function $H: \mathbb{R}^d \to \mathbb{R}$ as

$$H(x) = \mathbb{1}_{[0,\infty) \times \mathbb{R}^{d-1}}(x)$$

$$\mathcal{H}^\alpha[-\tfrac{1}{2}, \tfrac{1}{2}]^d = \{ \tilde{G} \circ T \in L^\infty[-\tfrac{1}{2}, \tfrac{1}{2}]^d : \tilde{G}(x) = H(x(1) + \phi(x(2), \dots, x(d)), x(2), \dots, x(d)),\ \phi \in C^\alpha(\mathbb{R}^{d-1}),\ T \text{ is a permutation matrix} \}$$

For horizon functions $G \in \mathcal{H}^\alpha(\mathbb{R}^d)$ we define the "norm" of G as

$$\| G \|_{\mathcal{H}^\alpha[-\frac{1}{2}, \frac{1}{2}]^d} = \| \phi \|_{C^\alpha[-\frac{1}{2}, \frac{1}{2}]^d}$$

Step 2 : Approximate horizon functions w/ ReLU networks

Lemma : Let $G \in \mathcal{H}^\alpha[-\frac{1}{2}, \frac{1}{2}]^d$ w/ $\| G \|_{\mathcal{H}^\alpha[-\frac{1}{2}, \frac{1}{2}]^d} = B < \infty$ and let $0 < \varepsilon < 1/2$. Then there are constants $t = t(d, \alpha, B) \in \mathbb{N}$ and $c = c(d, \alpha, B) > 0$ for which there is a neural network $g(x; \theta)$ with

$$L \leq (2 + \lceil \log_2 \alpha \rceil)(14 + 2\alpha/d)$$

layers and

$$\| \theta \|_0 \leq c \cdot \varepsilon^{-2(d-1)/\alpha}$$

nonzero, (t, ε)-quantized weights such that

$$\| G - g \|_{L^2[-\frac{1}{2}, \frac{1}{2}]^d} < \varepsilon \quad \text{and} \quad 0 \leq g(x; \theta) \leq 1$$

Idea of proof: Horizon functions are similar to cutoff functions, except instead of the jump discontinuity being a hyperplane (as in a cutoff function), it is along a hypersurface that is the graph of a C^α function. We have "proven" two results that can resolve this:

(1) NNs can approximate cutoff functions
(2) NNs can approximate C^α functions

Combine these two results to prove the lemma.

Step 3: Reduction to one set K w/ $\partial K \in C^\alpha$.

Note if we can approximate $\mathbb{1}_K(x)$, then we can approximate

$$F(x) = \sum_{m=1}^{M} a_m \mathbb{1}_{K_m}(x)$$

up to a constant depending on M.

Step 4: Introduce a space of sets based on horizon functions

$$\mathcal{K}^\alpha_{r,B}[-\tfrac{1}{2},\tfrac{1}{2}]^d = \{ K \subset [-\tfrac{1}{2},\tfrac{1}{2}]^d :$$

For all $x \in [-\frac{1}{2},\frac{1}{2}]^d$, there exists a $G_x \in \mathcal{H}^\alpha[-\frac{1}{2},\frac{1}{2}]^d$ with $\|G_x\|_{\mathcal{H}^\alpha[-\frac{1}{2},\frac{1}{2}]^d} \leq B$ for which

$$\mathbb{1}_K(x') = G_x(x')$$

for all $x' \in [-\frac{1}{2},\frac{1}{2}]^d$ with $\|x - x'\|_\infty \leq 2^{-r}$ $\}$

Remark: If $K \subset [-\frac{1}{2},\frac{1}{2}]^d$ with $\partial K \in C^\alpha$, then $K \in \mathcal{K}^\alpha_{r,B}[-\frac{1}{2},\frac{1}{2}]^d$ for some large enough r and B.

Step 5: Approximate $\mathbb{1}_K(x)$, $K \in \mathcal{K}^\alpha_{r,B}[-\frac{1}{2},\frac{1}{2}]^d$, with a ReLU neural network

Theorem: Let $K \in \mathcal{K}^\alpha_{r,B}[-\frac{1}{2},\frac{1}{2}]^d$ and $0 < \varepsilon < 1/2$. Then there are constants $t = t(d,r,\alpha,B) \in \mathbb{N}$ and $c = c(d,r,\alpha,B) > 0$ for which there is a neural network $f(x;\theta)$ with

$$L \leq (3 + \lceil \log_2 \alpha \rceil)(11 + 2\alpha/d)$$

layers and

$$\|\theta\|_0 \leq c \cdot \varepsilon^{-2(d-1)/\alpha}$$

nonzero, (t,ε)-quantized weights such that

$$\|\mathbb{1}_K - f\|_{L^2[-\frac{1}{2},\frac{1}{2}]^d} < \varepsilon$$

and $\|f\|_\infty \leq 1$

Idea of Proof: Any $K \in \mathcal{K}^\alpha_{r,B}[-\frac{1}{2},\frac{1}{2}]^d$ is locally a horizon function, which we can approximate by the lemma. Now patch together these local approximations.

Now let us consider lower bounds on the required complexity of NNs with (t, ε)-quantized weights for approximating horizon functions.

Note: horizon functions $\subset$ piecewise constant functions
$\subset$ piecewise smooth functions

So the results apply to these classes of functions as well

I am not going to state these results as precisely as the upper bound results.

We will consider two types of complexity:
(1) the number of nonzero weights, $\|\theta\|_0$
(2) the number of layers, L

We will see the results of the previous theorem are nearly sharp, meaning that one cannot do (much) better than the bound for $\|\theta\|_0$ and the bound for L.

On the number of nonzero weights: $\leftarrow \sigma(0)=0$ all that is required here

In order to guarantee, for any horizon function $G \in \mathcal{H}^\alpha[-\frac{1}{2}, \frac{1}{2}]^d$ with $\|G\|_{\mathcal{H}^\alpha[-\frac{1}{2},\frac{1}{2}]^d} \leq B$, that

$$\|G - g\|_{L^2[-\frac{1}{2},\frac{1}{2}]^d} \leq \varepsilon$$

for some neural network $g(x;\theta)$, the number of nonzero, (t, ε)-quantized weights must be at least

$$\|\theta\|_0 \geq \frac{c \cdot \varepsilon^{-2(d-1)/\alpha}}{\log_2(1/\varepsilon)}$$

where $c = c(d, \alpha, B)$ and $t = t(d, \alpha, B)$.

Remarks: (i) The neural network from the approximation theorem is nearly optimal since it had

$$\|\theta\|_0 \leq C \cdot \varepsilon^{-2(d-1)/\alpha}$$

nonzero, (t, ε)-quantized weights.

(ii) One can extend the result of Pinkus and Maiorov (1999) to horizon functions. Since the number of weights in the Pinkus/Maiorov network is fixed, regardless of ε, this result would seem to indicate they must either be very complex (i.e., not quantized) or very large. There is the technical point, though, that the nonlinearity of Pinkus/Maiorov does not satisfy $\sigma(0)=0$.

On the depth of the network:

Not precise version:
The number of layers must be at least

$$L \geq \frac{\alpha}{4(d-1)}$$

Thus, again, the number of layers in the approximation theorem,

$$L \leq \left(3 + \lceil \log_2 \alpha \rceil\right)\left(11 + \frac{2\alpha}{d}\right)$$

is nearly optimal.

Curse of dimensionality

Suppose $F = G \circ \tau$, where $\tau : \mathbb{R}^d \to \mathbb{R}^k$ is smooth and

$$G(z) = \sum_{i=1}^{M} a_m \mathbb{1}_{K_m}(z)$$

If τ is nice enough (e.g., $\tau \in C^\infty$, $D\tau(x)$ is full rank, plus some other properties) then one can prove something like:

For each $0 < \varepsilon < 1/2$ there is a neural network $f(x;\theta)$ with at most

$$\|\theta\|_0 \leq c \cdot \varepsilon^{-2(k-1)/\alpha}$$

nonzero, (t, ε)-quantized weights such that

$$\|F - f\|_{L^2[-\frac{1}{2},\frac{1}{2}]^d} < \varepsilon$$

Remarks: (i) c depends on d, but the rate $O(\varepsilon^{-2(k-1)/\alpha})$ depends on k

(ii) Number of layers is very large, and probably is suboptimal.