1 Model Framework

We consider continuous functions $f : D \mapsto I$, where $D \subseteq \mathbb{R}^m,\ I \subseteq \mathbb{R}^n$, that are implemented by neural networks. We mostly consider feed-forward networks which implicitly define a mapping from their input neurons to their output neurons.

It might feel strange to analyze feed-forward networks (in contrast to recurrent ones), since, intuitively, the human brain operates recurrently. However, there are multiple reasons for this choice: First, shortcomings of this architectural design help us understand beneficial design patterns. Second, while the entire human brain might operate recurrently, there still might be non-recurrent sub networks. Third, it is a known result from computer science that multilayer perceptrons are universal approximators (if there aren't any parameter constraints).

Theorem 1.1 (MLPs are Universal Approximators)**.** *Given any continuous function $f : D \mapsto I$, where $D \subseteq \mathbb{R}^m,\ I \subseteq \mathbb{R}^n$, and D is compact, there exists a multilayer perceptron that defines a function $f' : D \mapsto I$ s.t.*

$$\max_{\boldsymbol{x} \in D} \| f(\boldsymbol{x}) - f'(\boldsymbol{x}) \| < \epsilon$$

for any $\epsilon > 0$.

Clearly, based on the assumption we have to define an encoding of the instances of our problem domain to $\mathbb{R}^n$. This encoding is also invariant in time and space, which must not necessarily be the case in human brains. If you think of the number 1, your brain might show different activity at different times (maybe one time you are more stressed out; or you think differently of the number based on your current activity, as 1 also represents the multiplicative identity for example). Still, there might be some sub-circuitry activated when adding numbers where a certain set of neurons show similar activation patterns when adding by 1.

Now, there are rather boring mappings, like $f : \mathbb{N} \mapsto \{0, 1\}, f(n) \mapsto 1[n \text{ is even}]$. Seemingly, we humans can compute this functions over the entire set $\mathbb{N}$. However, I doubt this. If I asked you whether 985094890282377640950293869O2 is even or odd, you wouldn't have any issues. But if I asked you to repeat this number instead, most of us would fail (even if you managed to do this, what about even bigger and bigger numbers?).

Why did I claim this mapping to be boring? Because this problem can be reduced to a mapping $f' : \{0, 1, \ldots, 9\} \mapsto \{0, 1\}$ (by just considering the last digit). This mapping has a finite domain, and hence also a finite image. Such mappings can be implemented by simple input-output associations.