Mathematics for New Technologies in Finance

Exercise sheet 1

Exercise 1.1

(a) What is the formal definition of shallow or deep neural networks? (You might need the words affine function and activation function in the definition)

(b) Why we always like to consider non-linear activation functions?

(c) If the activation functions of neural network are bounded, e.g. $x \mapsto \tanh(x)$, is the neural network bounded for all possible inputs? If the activation functions are unbounded, e.g. ReLU, is the neural network bounded on compact input spaces?

(d) Is the sum of two neural networks still a neural network? Is the product of two neural networks still a neural network? Is the composition of two neural networks still a neural network?

(e) Are neural networks with ReLU activation functions differentiable?

Exercise 1.2

(a) Build a tent neural network $h\colon \mathbb{R} \to \mathbb{R}$ s.t.

$$h(x) = \begin{cases} x+1, & x \in [-1,0] \\ 1-x, & x \in [0,1] \\ 0, & \text{otherwise} \end{cases}, \tag{1}$$

from a ReLU neural network.

(b) Use (a) to prove the universal approximation theory on $[0,1]$ i.e. every continuous function f on [0, 1] can be uniformly approximated by neural networks (see the Faber-Schauder expansion of a continuous function).

Exercise 1.3 See notebook 1.

(a) Code a neural network to approximate a function on $[-5,5]$.

(b) Code a tent neural network on $\mathbb{R}^2$.

Exercise 1.4 Prove the real continuous function on compact set with supremum norm i.e. $(C(K), \|\cdot\|_\infty)$ is a Banach space.

Exercise 1.5 In the Black–Scholes model, the price of a European call option is given as follows:

$$C(X,r,\sigma,T,K) = X\Phi(d_1) - Ke^{-rT}\Phi(d_2), \text{ where}$$
$$d_1 = \frac{\ln(X/K) + (r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}$$
$$d_2 = d_1 - \sigma\sqrt{T}.$$

In the above, X is the current price of the underlying asset, r the interest rate, σ the volatility of the market, T the maturity, K the strike and Φ is the cumulative distribution function (CDF) of the standard normal distribution. We shall discuss this model in detail in another lecture. As for now, fix $X = 2$, $r = 0$, $T = 1$ and $K = 1$. In this way, the output $C = C(\sigma)$ only depends on the volatility σ, which we assume is taking values in $[0,1]$.

(a) See notebook 2. Code a feed-forward neural network to approximate this function, while varying the dimension of the training set.

(b) See notebook 2. Fix the elements in the training set to 100. Check how the number of layers in the network affects the output.

(c) What is the meaning of the parameter σ? What happens when $\sigma = 0$?