MNTF Mathematics for New Technologies in Finance

professor : Josef Teichmann

author : walkerchi

Approximation

Weierstrass

Weierstrass Approximation Theorem

A is dense in $C(\mathcal{X},\mathbb{R}^m)=\{f_i|f_i\in C^0_{pw},f_i\in\mathcal{X}\to\mathbb{R}^m,\mathcal{X}\subset\mathbb{R}^n\}$ if

1. A contains all polynomial functions: $\mathcal{P}\subset A$
    1. A is vector subspace of C : $A\subset C(\mathcal{X},\mathbb{R}^m)$
        - $f_1(x)+f_2(x)=f_3(x)\quad\forall f_1,f_2\in A,\exists f_3\in A$
        - $cf_1(x)=f_2(x)\quad\forall c\in\mathbb{R},\forall f_1\in A,\exists f_2\in A$
    2. A is closed under multiplication : $f_1(x)f_2(x)=f_3(x)\quad\forall f_1,f_2\in A,\exists f_3\in A$
    3. A contains constant function : $f(v)=c\quad\forall v\in\mathcal{X},\exists f\in A$
2. points seperation : $f(v)\neq f(w)\quad\forall v\neq w\wedge v,w\in\mathcal{X}$

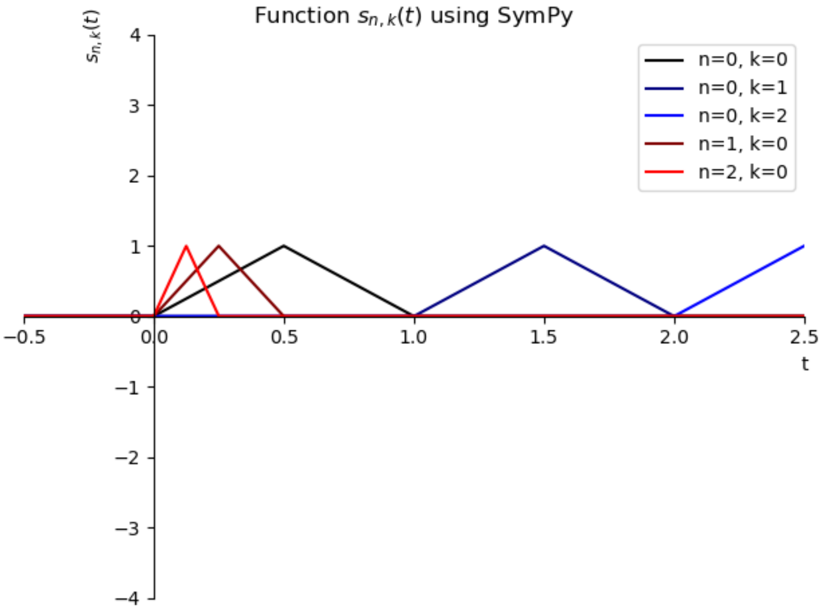
for shallow NN with ReLU

- [ ] contains all polynomial functions
    - [ ] vector space
    - [ ] closed under multiplication
    - [x] contains constant function
- [x] points seperation

$\Rightarrow$ NN with ReLU is dense in $C(\mathcal{X},\mathbb{R}^m)$

Faber-Schauder

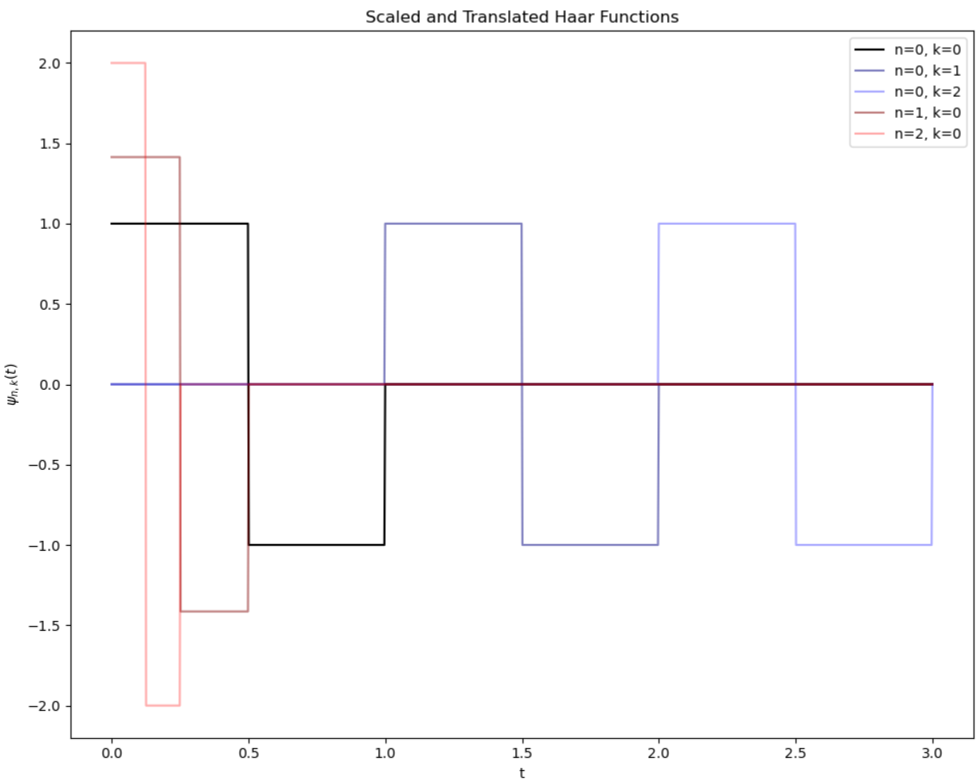
Faber-Schauder basis : $s_{n,k}=2^{1+\frac{n}{2}}\int_0^t\psi_{n,k}(u)\mathrm{d}u\quad n,k\in\mathbb{Z}$



$$v = \sum_{n=0}^{\infty} \alpha_n b_n \quad \forall v \in \mathbb{R}, \exists \alpha_n \in \mathbb{R}, \exists b_n \in \{s_{*,*}\}$$

- equivalent to the linear combination of ReLU

Haar function : $\psi_{n,k}(t) = 2^{n/2}\psi(2^n t - k) \quad n, k \in \mathbb{Z} \quad \psi(t) = \begin{cases} 1 & t \in [0, \frac{1}{2}) \\ -1 & t \in [\frac{1}{2}, 1) \\ 0 & \text{otherwise} \end{cases}$



- $\operatorname{supp}(\psi_{n,k}) = [k2^{-n}, (k+1)2^{-n})$
- $\int_{\mathbb{R}} \psi_{n,k}(t) \mathrm{d}t = 0$
- $\|\psi_{n,k}\|_{L^2(\mathbb{R})} = 1$
- $\int_{\mathbb{R}} \psi_{n_1,k_1} \psi_{n_2,k_2} \mathrm{d}t = \delta_{n_1 n_2} \delta_{k_1 k_2}$

Banach

A is Banach space if :

1. Cauchy sequence : $\|f_m - f_n\| \leq \epsilon \quad \forall \epsilon > 0, \exists N_\epsilon \in \mathbb{N}, \forall m, n > N_\epsilon$
2. completeness : $\|f - f_m\| \leq \epsilon \quad \forall \epsilon > 0, m \to \infty$

Signature

for path/curve $X_t \in \mathbb{R}^d$, $X_t = \begin{bmatrix} X^1(t) & X^2(t) & \cdots & X^d(t) \end{bmatrix}^\top$, signature could determine the curve in tree like equivalences

n-th level of signature : $S(X)_{a,b}^{i_1,i_2,\cdots,i_n} = \int_a^b \int_a^{t_{n-1}} \cdots \int_a^{t_2} \mathrm{d}X_{t_1}^{i_1} \cdots \mathrm{d}X_{t_n}^{i_n} \quad n \leq d$

- $S(X)_{a,b}^{i_1,i_2,\cdots,i_n} \in \mathbb{R}^{n^{\otimes d}}$, it's a d dimension tensor with each dimension of span n

signature : $S(X)_{a,b} = (1, \mathrm{Sig}_{[a,b]}(X)^{i_1}, \mathrm{Sig}_{[a,b]}(X)^{i_1,i_2}, \cdots, \mathrm{Sig}_{[a,b]}^{i_1,i_2,\cdots,i_d})$

- the maximum length of $S(X)_{a,b}$ is $d^0 + d^1 + d^2 + \cdots + d^d = \frac{d^{d+1}-1}{d-1}$
- the length of depth M is $d^0 + \cdots + d^M = \frac{d^{M+1}-1}{d-1}$

normally we got :

- $S(X)_{a,b}^i = \int_a^b dX = X_b^i - X_a^i$
- $S(X)_{a,b}^{i,j} = \int_a^b \int_a^{t_2} dX_{t_1}^i dX_{t_2}^j = \int_a^b (X_{t_2}^i - X_a^i) dX_{t_2}^j \overset{X=\alpha t+\beta}{=} \frac{1}{2}(X_b^i - X_a^i)(X_b^j - X_a^j)$
- $S(X)_{a,b}^{i,j,k} = \int_a^b \int_a^{t_3} \int_a^{t_2} dX_{t_1}^i dX_{t_2}^j dX_{t_3}^k \overset{X=\alpha t+\beta}{=} \frac{1}{6}(X_b^i - X_a^i)(X_b^j - X_a^j)(X_b^k - X_a^k)$
- shuffle product rule : $S(X)_{a,b}^I S(X)_{a,b}^J = \sum_{K=\mathrm{shuff}([I_1,\cdots,J_1,\cdots])} S(X)_{a,b}^K$
    - example : $S(X)_{a,b}^1 S(X)_{a,b}^2 = S(X)_{a,b}^{1,2} + S(X)_{a,b}^{2,1}$

Financial Market

- S_t^i : i-th asset prices at time t, $S \in \mathbb{R}^{N \times d+1}$, normally S^0 represent bank account
- ϕ_t^i : holdings in i-th assets at time t
- V_t : value of portfolio at time t , $V_t = \sum_i \phi_t^i S_t^i$

self-financing : $\sum_i \phi_{t+1}^i S_t^i = \sum_i \phi_t^i S_t^i \quad \forall t \in [0, N)$

value process : $V_{t+1} - V_t = \sum_i \phi_t^i (S_{t+1}^i - S_t^i) \quad \forall t \in [0, N)$

martingale : $\mathbb{E}[X_{n+1} | X_1, \ldots, X_n] = X_n$

arbitrage : $\underbrace{P(V_t \geq 0) = 1}_{\text{no risk of losing money}} \wedge \underbrace{P(V_t \neq 0) > 0}_{\text{portfolio value} > 0} \quad t \in (0, T), \qquad \underbrace{V_0 = 0}_{\text{requires no initial value}}$

Stochastic Differential Equation

brownian motion/wiener process : $W_{t+1} - W_t \sim \mathcal{N}(0,1) \quad W_0 = 0$

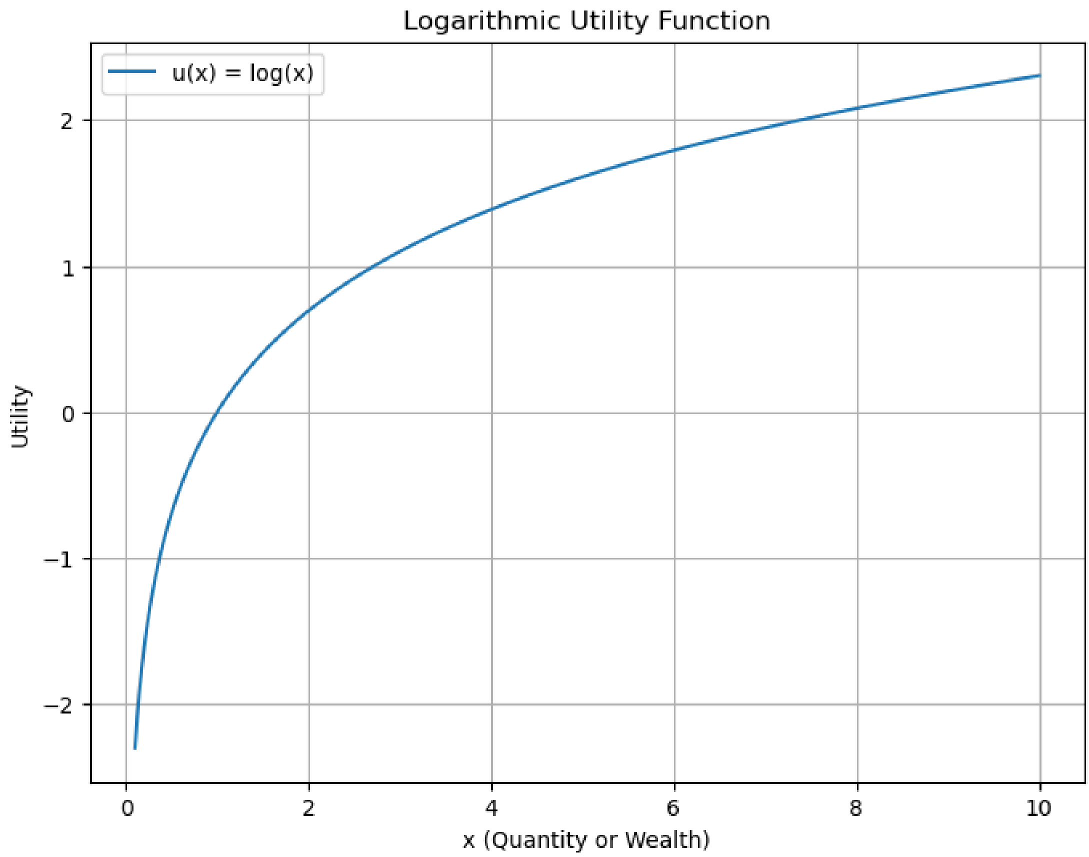
Geometric Brownian motion : $dS_t = \mu\ S_t\ dt + \sigma\ S_t\ dW_t \Leftrightarrow S_t = S_0 e^{\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t}$

- W_t is brownian motion/wiener process
- μ, σ is the expectation/variance for the GBM

Utility

utility function u : the additional utility or satisfaction from consuming one more unit of a good decreases as more of the good is consumed.

- concave : $f''(x) < 0$
- monotone increase : $f'(x) > 0$



expected utility optimization problem : $\underset{\phi_t^i}{\operatorname{argmax}}\ E[u(V_N)]$

Local Volatility Model : $\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma(S_t, t)S_t\mathrm{d}W_t$

- S_t : underlying asset price at time t
- r risk-free interesting rate
- $\sigma(S_t, t)$: local volatility function
- W_t is the Brownian motion/ Wiener process

Local Stochastic volatility model : $\begin{aligned} \mathrm{d}S_t &= \mu S_t \mathrm{d}t + \sqrt{\nu_t} S_t \mathrm{d}W_t \\ \mathrm{d}\nu_t &= \alpha_t(\nu)\mathrm{d}t + \beta_t(\nu)\mathrm{d}W_t' \end{aligned}$

- $\alpha_t(\nu), \beta_t(\nu)$: functions based on ν
- W_t, W_t' : Wiener process with correlation factor ρ
- ν_t : model the variance of S_t, it relies on another stochastic process, so LSV is not a standard SDE

Heston model : $\mathrm{d}\nu_t = \kappa(\theta - \nu_t)\mathrm{d}t + \xi\sqrt{\nu_t}\mathrm{d}W_t'$

- θ : long term variance
- κ : rate of variance reverts toward it's long term
- ξ : volatility of volatility, the variance of ν_t

Ito's lemma : $\mathrm{d}f(S,t) = \left(\frac{\partial f}{\partial t} + \mu\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2\frac{\partial^2 f}{\partial S^2}\right)\mathrm{d}t + \sigma\frac{\partial f}{\partial S}\mathrm{d}W_t \quad \mathrm{d}S(t) = \mu\mathrm{d}t + \sigma\mathrm{d}W_t$

Black Scholes equation : $\frac{\partial C}{\partial t} + rK\frac{\partial C}{\partial K} + \frac{1}{2}\sigma^2K^2\frac{\partial^2 C}{\partial K^2} - rC = 0$: derive from Ito's lemma

- $C(K,t)$: European call option price, equivalent to value V
- K : strike price, equivalent to assets/stock price S

Dupire's formula : $-\frac{\partial C}{\partial T} - rK\frac{\partial C}{\partial K} + \frac{1}{2}\sigma^2K^2\frac{\partial^2 C}{\partial K^2} - \Delta C = 0$

- when $r = 0$, $\sigma^2 = \frac{2\partial_T C}{K^2\partial_K^2 C}$

Breeden-Litzenberger fromula : $\partial_K^2 C(T,K)\mathrm{d}K = p_T(K)\mathrm{d}K$

- $p_T(K)\mathrm{d}K$ is the risk neural probability, $p_T(K) = p(S_t \in [K, K+\mathrm{d}K])$

Deep portfolio optimization

$$\mathrm{d}S_t = S_t\mu\mathrm{d}t + S_t\sigma\mathrm{d}W_t$$
$$\mathrm{d}X_t = \alpha_t X_t\frac{\mathrm{d}S_t}{S_t} + (1-\alpha)X_t r\mathrm{d}t$$
$$\max_\alpha \mathbb{E}[u(X_T)]$$

- X_t is the money at time t
- α_t is strategy how much portion of money in the stock rather than in the bank at time t
- S_t is the stocks prices, governed by parameter μ and σ , with W_t a brownian motion or wiener process
- r is the interest rate saved in bank
- u is the utility function ,normally $u(x) = \frac{x^\gamma - 1}{\gamma}$

analytical solution : $\alpha^* = \frac{\mu - r}{\sigma^2(1-\gamma)}$

Deep Hedging

$$\mathrm{d}S_t = S_t\mu\mathrm{d}t + S_t\sigma\mathrm{d}W_t$$

$$\underset{H,\pi}{\operatorname{argmin}}\ \mathbb{E}\left[\left\| f(S_T) - \pi - \int_0^T H_t \mathrm{d}S_t \right\|^2\right]$$

- S_t is the risky stocks prices, governed by parameter μ and σ , with W_t a brownian motion or wiener process
- $f(S_t)$ is financial claim, the payoff is $f(S_T) = \max(S_T - K, 0)$ for European call, K is the strike price
- π the price of the option, the upfront payment you received
- H_t is the hedge strategy at time t
- T is the expire date

Deep Calibration

Heston Calibration

$$\mathrm{d}X_t = \left((q-r) - \frac{1}{2}Y_t\right)\mathrm{d}t + \sqrt{Y_t}\mathrm{d}W_t^1$$

$$\mathrm{d}Y_t = (\theta - \kappa Y_t)\mathrm{d}t + \sigma\sqrt{Y_t}\mathrm{d}W_t^2$$

$$\underset{\theta,\kappa,\sigma}{\operatorname{argmin}} \sum_{t=0}^{T} \|X_t - \log(S_t)\|^2$$

- r : interest rate
- q : dividend
- S_t : price of assets
- X_t : predicted log price : $X_0 = \log(S_0)$
- Y_t : variance of Heston model : $Y_0 = \nu_0$

Utility Calibration

$$dS_t = S_t\alpha_t l(t, S_t)\mathrm{d}W_t$$

$$\underset{l}{\operatorname{argmin}} \left\| \mathbb{E}\left[\max(S_T - K, 0) - C(K,T) - \int_0^T H_t \mathrm{d}S_t\right] \right\|^2$$

- α_t is exogenous process at time
- $l(t, S_t)$ is leverage function
- S_t is the stocks prices, with W_t a brownian motion or wiener process
- H_t is the hedge strategy at time t
- K is the strike price of European call
- C is the European call option market price

Deep Simulation

model controlled differential equation

$$\mathrm{d}X_t = \sum_{i=0}^{d} \sigma(A_i X_t + b_i)\mathrm{d}u_i(t)$$

- A_i, b_i are randomly generated matrices/vectors
- u_i is control coefficient learned by network

Reinforcement Learning

- a, s : action $a \in A$, state $s \in S$
- V, V^* : value function, optimal value function, $V \in S \times T \to \mathbb{R}$
- $\pi(s)$: policy , $\pi \in S \to A$
- $c(t, s, a)$: cost function , $c \in T \times S \times A \to \mathbb{R}$
- $r, R(s, a)$: reward, reward function , $r \in \mathbb{R}, R \in S \times A \to \mathbb{R}$
- $Q(s, a)$: Q/state action function, return the priority for each state and action, $Q \in S \times A \to \mathbb{R}$

[DPP]Dynamic programming principle :
$V^*(t, s) = \max_a \left\{ \int_t^T c(\tau, s(\tau), a(\tau))\mathrm{d}\tau + V*(T, s(T)) \right\}$

[HJB] Hamiton-Jacobi-Bellman equation : $\frac{\partial V(s,t)}{\partial t} + \max_a \left(\frac{\partial V(s,t)}{\partial s} \cdot f(t, s, a) + c(t, s, a) \right) = 0$

- $f(t, s, a)$: system dynamics, how state change over time, $\frac{\mathrm{d}s(t)}{\mathrm{d}t} = f(t, s, a)$

Bellman equation : $Q(s, a) = r + \gamma \max_{a'} Q(s', a')$

Value Iteration : $V^{(n+1)} = \max_a \left\{ R(s, a) + \gamma \sum_{s'} P(s|s', a) V^{(n)}(s') \right\}$

$$V^{\pi^{(n)}}(s) = R(s, \pi(s)) + \gamma \sum_{s'} P(s'|s, \pi(s)) V^{\pi^{(n)}}(s')$$

Policy Iteration :

$$\pi^{(n+1)} = \underset{\pi}{\operatorname{argmax}} \left\{ R(s, a) + \gamma \sum_{s'} P(s'|s, a) V^{\pi^{(n)}}(s') \right\}$$

Q learning*(environment-known/model-based)* :

$Q(s, a) \leftarrow R(s, a) + \sum_{s'} P(s'|s, a) \left[\gamma \max_{a'} Q(s', a') \right]$

Q learning*(environment-unknown/model-free)* :

$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$

Optimization

inverse calibration : $\underset{\theta}{\operatorname{argmin}} \|\mathbf{d} - \mathcal{NN}_\theta\|^2$

- $\mathbf{d}$ is the observed data
- $\mathcal{NN}_\theta \quad \theta \in \Theta$ is the pool of the model

optimization approach : $\underset{\theta}{\operatorname{argmin}} \|\mathbf{d} - \mathcal{NN}_\theta\|^2 + \lambda R_\theta$

- θ model parameters
- R_θ : regularization term ($|\cdot|$: lasso(L1) or $\|\cdot\|^2$: ridge(L2))

bayesian optimization :

$$P(M_i|\mathbf{d}) = \frac{P(\mathbf{d}|M_i)P(M_i)}{P(\mathbf{d})} \propto P(d|M_i)P(M_i)$$

- $P(M_i|\mathbf{d})$ posterior probability of model M_i given data $\mathbf{d}$
- $P(\mathbf{d}|M_i)$ likelihood of data given model M_i
- $P(M_i)$: prior probability of model M_i
- $P(\mathbf{d})$: evidence likelihood

for linear model $Y \sim \mathcal{N}(\theta X, \sigma^2\mathbf{I}), \theta \sim \mathcal{N}(0, \tau^2\mathbf{I})$, the maximizing posterior of $p(\theta|x, y)$ is ridge regression:

$$\begin{aligned}
\underset{\theta}{\operatorname{argmax}}\, p(\theta|x,y) &\propto \underset{\theta}{\operatorname{argmax}}\, p(\theta)p(y|x,\theta) \\
&\propto \underset{\theta}{\operatorname{argmax}}\, \exp\left(-\theta^\top \mathbf{I}\theta/\tau^2\right) \exp\left(-(y-\theta x)^\top \mathbf{I}(y-\theta x)/\sigma^2\right) \\
&\propto \underset{\theta}{\operatorname{argmin}}\, \frac{\sigma^2}{\tau^2}\|\theta\|^2 + \|y - \theta x\|^2
\end{aligned}$$

[SGLD]Stochastic Gradient Langevin Dynamics : gradient descent plus noise :

$$\mathrm{d}\theta_t = \frac{1}{2}\nabla \log p(\theta_t|x_1, \ldots, x_n)\mathrm{d}t + \mathrm{d}W_t$$

- escape from local minimal