# Applications of diffusion processes in Finance

Berkeley
UNIVERSITY OF CALIFORNIA

# Diffusion for generative learning (1/5)

- A successful generative learning model will satisfy the following -
  - High-quality sampling
  - Capture sample diversity
  - Fast generation of samples
- Current methods focus mainly on the high-quality sampling
- Capturing data diversity to avoid biases in learned models is highly desirable
- Eg: Cases where long-tailed distribution are important, we don't want to undersample in tails
- A need for faster sampling is required in applications for real-time usage

Berkeley
UNIVERSITY OF CALIFORNIA

# Diffusion for generative learning (2/5)

- Diffusion processes can be used as a substitute for GANs
- Models called Denoising diffusion models or score-based generative models give high quality samples, outperforming GANs and model data distribution in tails very well
- Denoising diffusion models consists of -
    - Forward diffusion - Maps data to noise(Gaussian) by sampling Gaussian noise and incrementally add it to the data
    - Reverse process - Undoes forward diffusion and iteratively denoises random noise into realistic data [Requires training of a NN here]

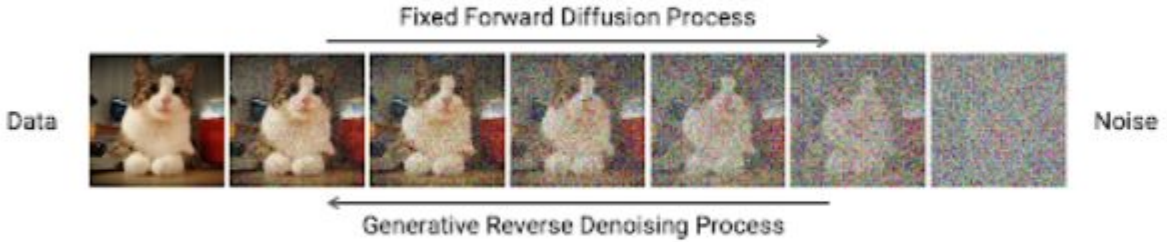
# Diffusion for Generative learning (3/5)

- Example -

Denoting a data point by $x_0$, and it's diffused version at timestep t by $x_t$, the forward process is defined as - $q(x_{1:T}|x_0) = \prod_{t\geq 1} q(x_t|x_{t-1}), \quad q(x_t|x_{t-1}) = N(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$

The reverse generative process is similarly defined in reverse order by -

$$p_\theta(x_{0:T}) = p(x_T) \prod_{t\geq 1} p_\theta(x_{t-1} \mid x_t), \quad p_\theta(x_{t-1} \mid x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 I)$$




Berkeley
UNIVERSITY OF CALIFORNIA

# Diffusion for generative learning (4/5)

- The discrete processes described on the slide before can be approximated by SDEs leading to continuous time diffusion models
- Working with SDEs is easier since-
  - More generic, can be converted to discrete time counterpart whenever needed
  - Can be solved using many available SDE solvers
  - Can be converted to ODEs (Dynkin’s formula) which are easy to work with
- However, solving the generative SDE is often complex and requires lot of computation for NN to generate a sample, hence GANs beat diffusion model in terms of speed

# Diffusion for Generative learning (5/5)

- NVIDIA research team proposed 3 solutions for speeding up generative process - Denoising diffusion GAN leads to massive speed up
- Denoising Diffusion GAN Idea -
  - Reduce the number of steps required for denoising in the reverse process
  - Observation - Learned denoising distribution in reverse synthesis process can be approximated by Gaussian distribution, but only for small denoising steps, leading to slow reversal
  - When large steps are used, a non-Gaussian multimodal distribution is needed
- Denoising Diffusion GANs represent denoising model using a multimodal conditional GAN, enabling to efficiently generate data in as few as 2 steps

Berkeley
UNIVERSITY OF CALIFORNIA

# Application in finance - sampling data

- Currently, we are using Quantlib's FDM for generating ground truth for training our model which takes a lot of time
- We can possibly, train a denoising diffusion GAN using few ground truth values and continuously sample from trained diffusion process to quickly sample more data points

# Policy regularization using diffusion models (1/2)

- Offline RL aims to learn policy on a static dataset and hence tends to perform poor on out of sample dataset
- There exist regularization methods for Offline RL algorithms but they tend to restrict policy space a lot often and hence lead to sub-optimal solutions
- Zhendong, Jonathan, Mingyuan explored policy regularization using diffusion processes
- They introduced Diffusion-QL, a new offline RL algorithm that leverages diffusion models to do precise policy regularization

# Policy regularization using diffusion models (2/2)

- Condition on state, an action pair is generated using conditional diffusion process
- A separate reward model is learned to predict the cumulative reward for each trajectory generated. This is then injected into reverse sampling stage to learn a trajectory given cumulative reward
- Benefits - g distribution matching technique and hence it could be seen as a powerful sample-based policy regularization method without the need for extra behavior cloning
- Applications - Portfolio construction and management, trading bots

Berkeley
UNIVERSITY OF CALIFORNIA

# Detecting change in drift of Brownian Motion (1/1)

- Shiryayev and Roberts proposed a diffusion process model (Shiryayvev-Roberts process) for detecting change in drift of Brownian motion
- Given a Brownian motion process W(t) which has some drift $\mu_1$ in time interval $[0,t_0]$ and drift $\mu_2$ for time $t > t_0$ , the model proposes a stopping rule T which detects the change point $t_0$ as soon as possible
- Application - A shock in market may change drift of Brownian motion for an asset and might be needed to be incorporated into pricing models or trading strategies to quickly adjust to the new market conditions

Berkeley
UNIVERSITY OF CALIFORNIA