Dynamical Systems in Machine Learning

Cédric Simal
23/02/2023

PSA

"A parody uncle sam poster with the message "I WANT YOU for SYSDYN", parody poster, digital painting, HQ, 4k", Made with Stable Diffusion

Motivation

The encroaching critical transition

ROBBIE THE WRITING ROBOT —

BuzzFeed preps AI-written content while CNET fumbles

200 percent BuzzFeed stock rise might signal start of a "pivot to AI" media trend.

BENJ EDWARDS - 1/27/2023, 7:10 PM

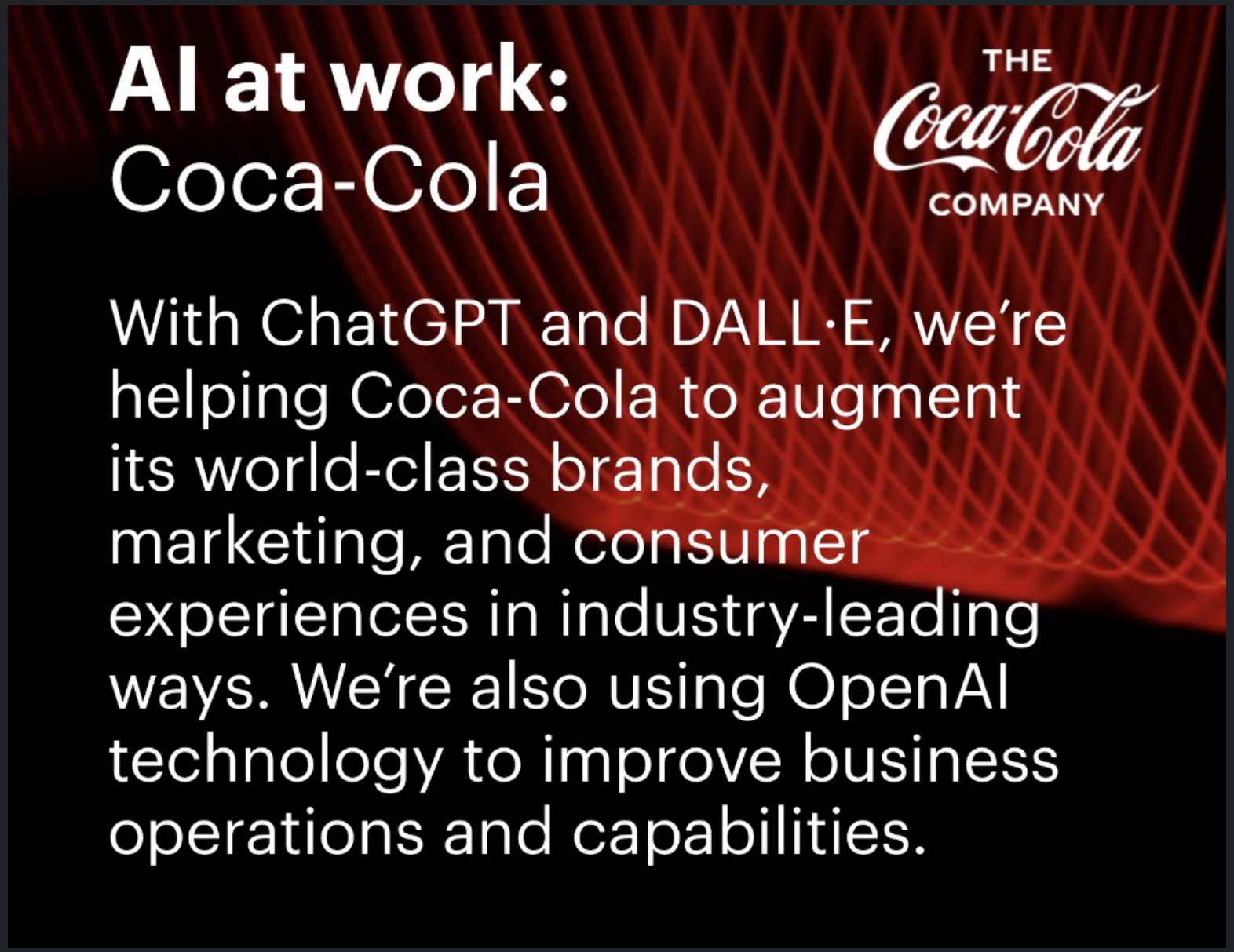
TECH • ARTIFICIAL INTELLIGENCE

The New AI-Powered Bing Is Threatening Users. That's No Laughing Matter

Lawsuits over Stability AI's Stable Diffusion could threaten the future of AI-generated art

https://arstechnica.com/information-technology/2023/01/pivot-to-chatgpt-buzzfeed-preps-for-ai-written-content-while-cnet-fumbles/

https://www.businessinsider.com/stable-diffusion-lawsuit-getty-images-stablility-ai-art-future-2023-1?op=1&r=US&IR=T



https://www.bain.com/vector-digital/partnerships-alliance-ecosystem/openai-alliance/

"Man-made horrors beyond your comprehension, Zdzisław Beksiński, The Coca Cola Company, Horror, Matte painting, HQ, 4k", made with Stable Diffusion

Today's Program

0. Basics of Neural Networks
1. Optimization Algorithms
2. Generative Adversarial Networks
3. Recurrent Neural Networks
4. ResNets and Neural ODEs
5. Diffusion Models

What's a Neural Network again?

$$\Phi(x) = (\varphi_L \circ \cdots \circ \varphi_1)(x)$$

↳ depth

Layers $\varphi_\ell(x_\ell) = \sigma_\ell \cdot (W_\ell x_\ell + b_\ell)$

activation function (applied pointwise)

weights $(\mathbb{R}^{n_{\ell+1} \times n_\ell})$ bias $(\mathbb{R}^{n_{\ell+1}})$

parameters θ

How to train your Neural Net

$$\min_{\theta} \underbrace{\mathcal{L}(x, y, \theta)}_{\text{Loss Function}} = \sum_{i \in I} L(y_i, \Phi_\theta(x_i))$$

↳ dataset $\{(x_i, y_i)\}_{i \in I}$

trained with Gradient Descent

$$\theta_{k+1} = \theta_k - \eta \nabla_{\theta_k} \mathcal{L}$$

↓

Learning Rate

Vignette 1: Optimization Algorithms

Pitfalls of training NNs

- For large datasets, evaluating the loss gets expensive
- The loss function is often non-convex
- No line search
- Training can take days, or months

Optimization Algorithms

Stochastic Gradient Descent

Replace the loss gradient by its expectation over the dataset

$$\theta_{k+1} = \theta_k - \eta \mathbb{E}_{i \in I}[\nabla_{\theta_k} L]$$

- **Stochastic** can get out of local minima
- Loss gets evaluated on random samples without replacement
- Still an ODE as $\eta \to 0$

Optimization Algorithms

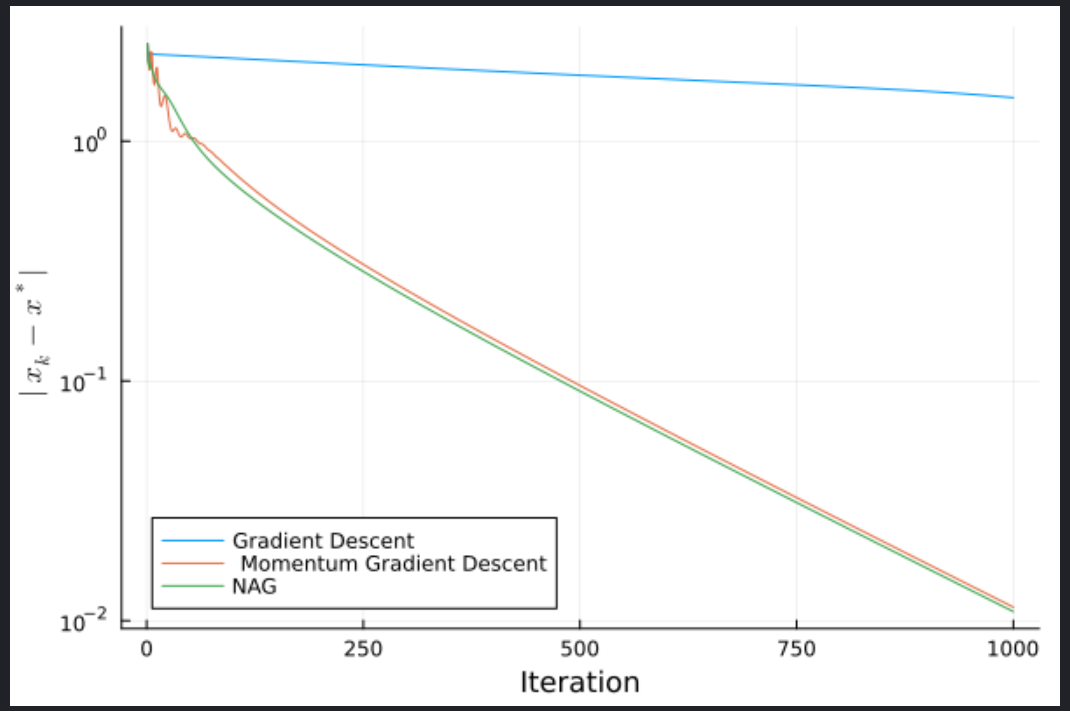
Momentum methods

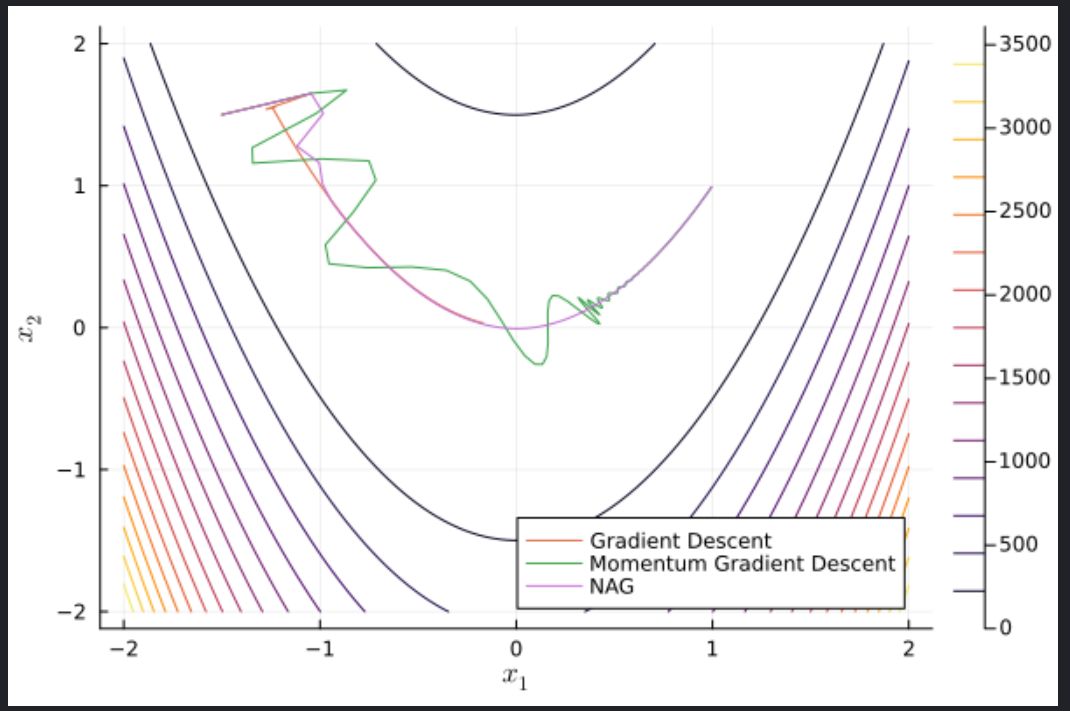
Add scaled past gradients

$$v_{k+1} = \gamma v_k + \eta \nabla_{\theta_k} \mathcal{L}$$
$$\theta_{k+1} = \theta_k - v_{k+1}$$

- Second order system with gradient as acceleration
- In the ODE limit, this is a Hamiltonian system [1]

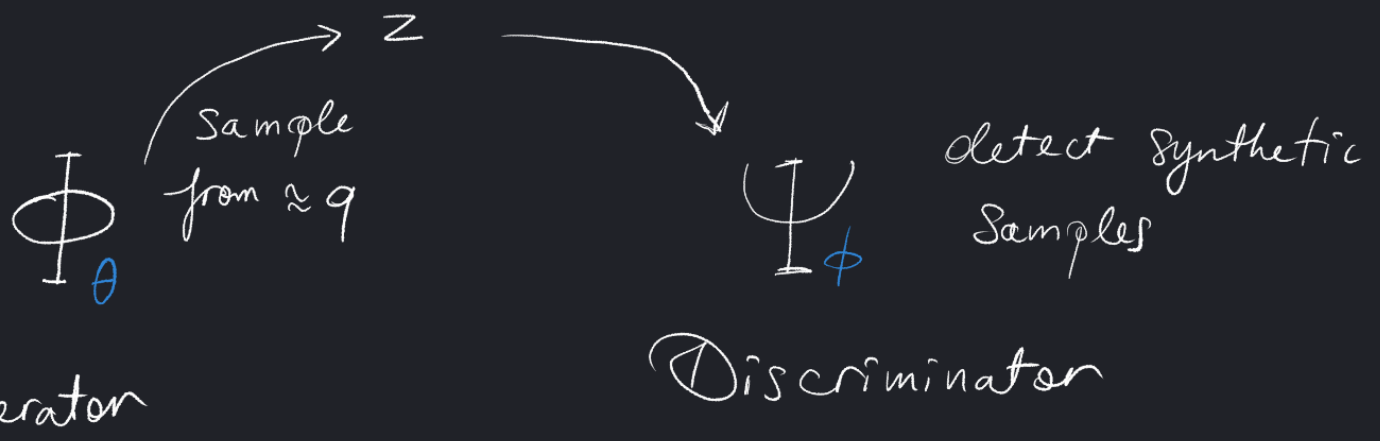
[1] J. Diakonikolas, M.I. Jordan, *Generalized Momentum-Based Methods: A Hamiltonian Perspective*, SIAM J. Optim., 31 (1), (2021)


10^0
10^{-1}
10^{-2}
$|x_k - x^*|$
0
250
500
750
1000
Iteration
Gradient Descent
Momentum Gradient Descent
NAG


2
1
0
−1
−2
x_2
−2
−1
0
1
2
x_1
3500
3000
2500
2000
1500
1000
500
0
Gradient Descent
Momentum Gradient Descent
NAG

Vignette 2: Generative Adversarial Networks

Dataset $X \sim q$ data distribution



Simultaneous Gradient Descent

$$\begin{cases} \theta_{k+1} = \theta_k - \eta \nabla_{\theta_k} f(\theta_k, \phi_k) \\ \phi_{k+1} = \phi_k - \eta \nabla_{\phi_k} g(\theta_k, \phi_k) \end{cases}$$

Not a Gradient Flow!

https://www.inference.vc/my-notes-on-the-numerics-of-gans/

Vignette 3: Recurrent Neural Networks

$$x_{n+1} = \phi(x_n)$$

$$= \phi^{\circ n}(x_0)$$

Problem : gradient $\nabla_\theta \mathcal{L}(x_n)$ is prone to

Converge to zero / diverge

$$\nabla_\theta \mathcal{L}(x_n) = \frac{\partial \mathcal{L}}{\partial x_n}\frac{\partial x_n}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial x_n}\left[\frac{\partial \phi}{\partial \theta} + \frac{\partial \phi}{\partial x_{n-1}} \cdot \frac{\partial x_{n-1}}{\partial \theta}\right]$$

⚠ Notation Abuse !

$$= \text{“}\frac{\partial \mathcal{L}}{\partial x}\left[\frac{\partial \phi}{\partial \theta}\sum_{k=0}^{n-1}\left[\frac{\partial \phi}{\partial x}\right]^k\right]\text{”}$$

↳ Jacobian

Vignette 4: Residual Networks

$$x_{\ell+1} = x_\ell + h\varphi_\ell(x_\ell)$$

Motivation: force φ_ℓ to stay "close" to identity
Alternative: Looks like Euler's method! $\rightarrow$ force φ_ℓ to be "regular"

Idea 1: Higher-order discretizations

Neural ODEs

Idea 2: $h \to 0$

$$\dot{x} = \Phi(x, t, \theta)$$

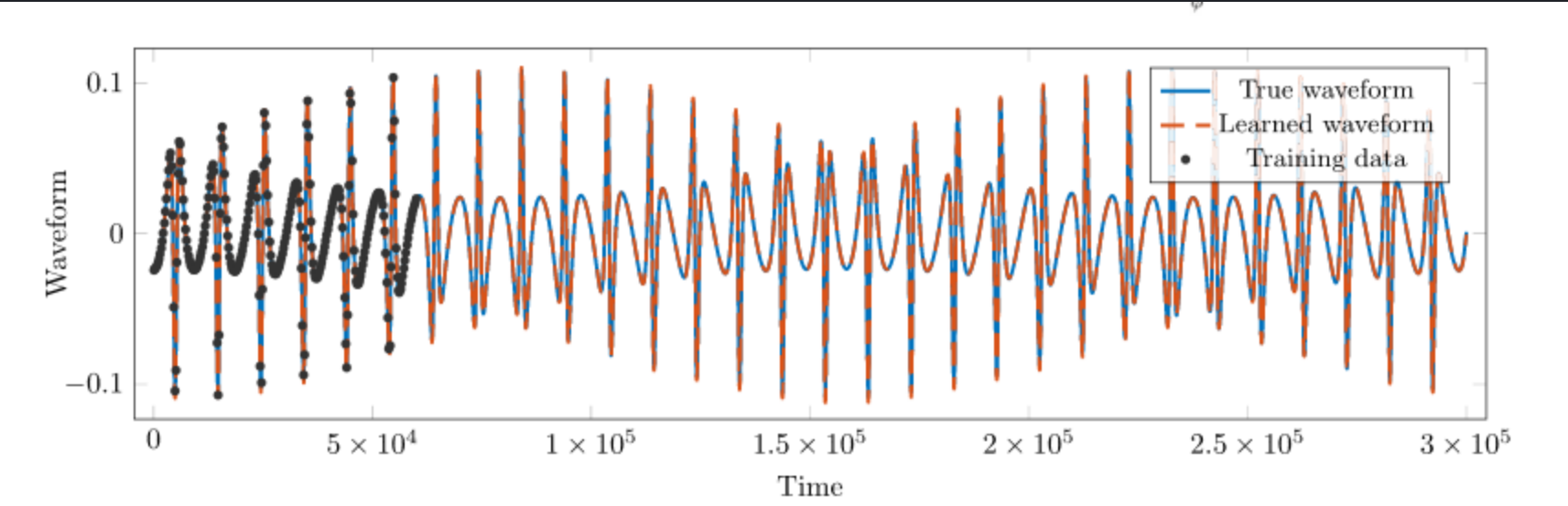
Advantages

- Can use black-box ODE solvers
- Adaptive time steps $\to$ Can handle irregularly sampled time series
- Controller design, Physical Modeling, ...
- Fit missing/unknown terms in existing scientific models (Scientific ML)

Open Access

Learning orbital dynamics of binary black hole systems from gravitational wave measurements

Brendan Keith, Akshay Khadse, and Scott E. Field
Phys. Rev. Research **3**, 043101 – Published 9 November 2021



https://phys.org/news/2021-11-machine-derive-black-hole-motion.html

Vignette 5: Diffusion Models

add noise

$\mathcal{N}(0, \sigma_t)$

add more noise

$\mathcal{N}(0, \sigma_t)$

Complicated distribution

(e.g. supported on an immersed manifold)

slightly less complicated distribution

$\approx$ Gaussian distribution

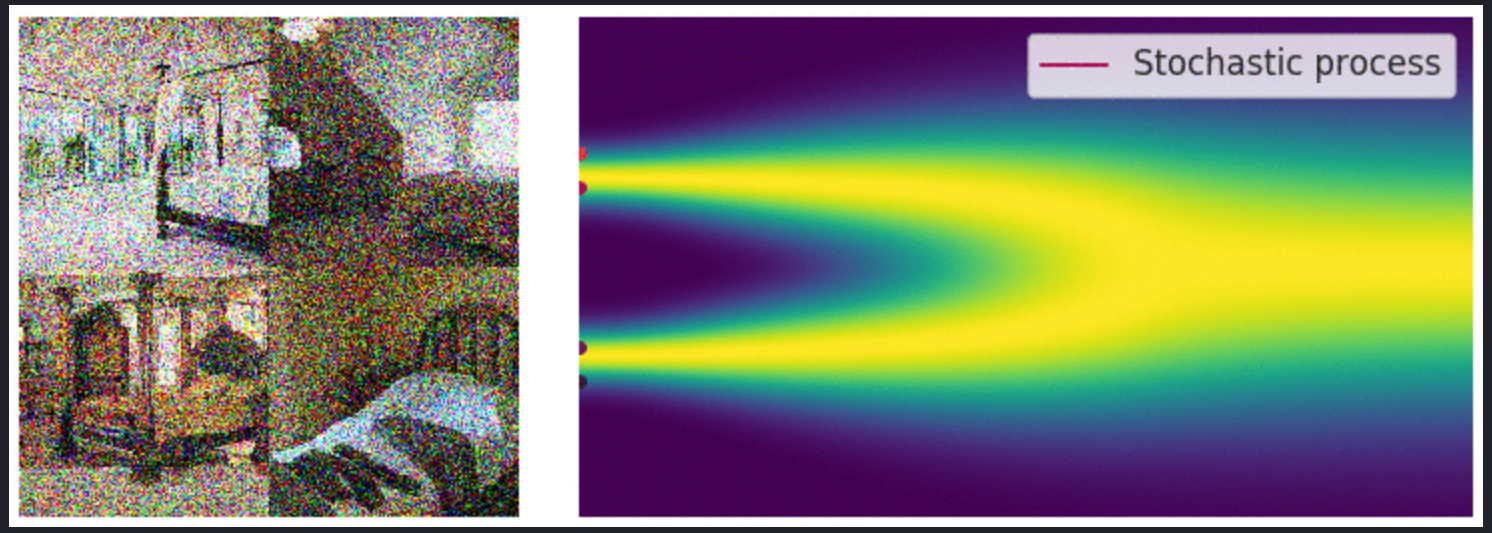
Forward process

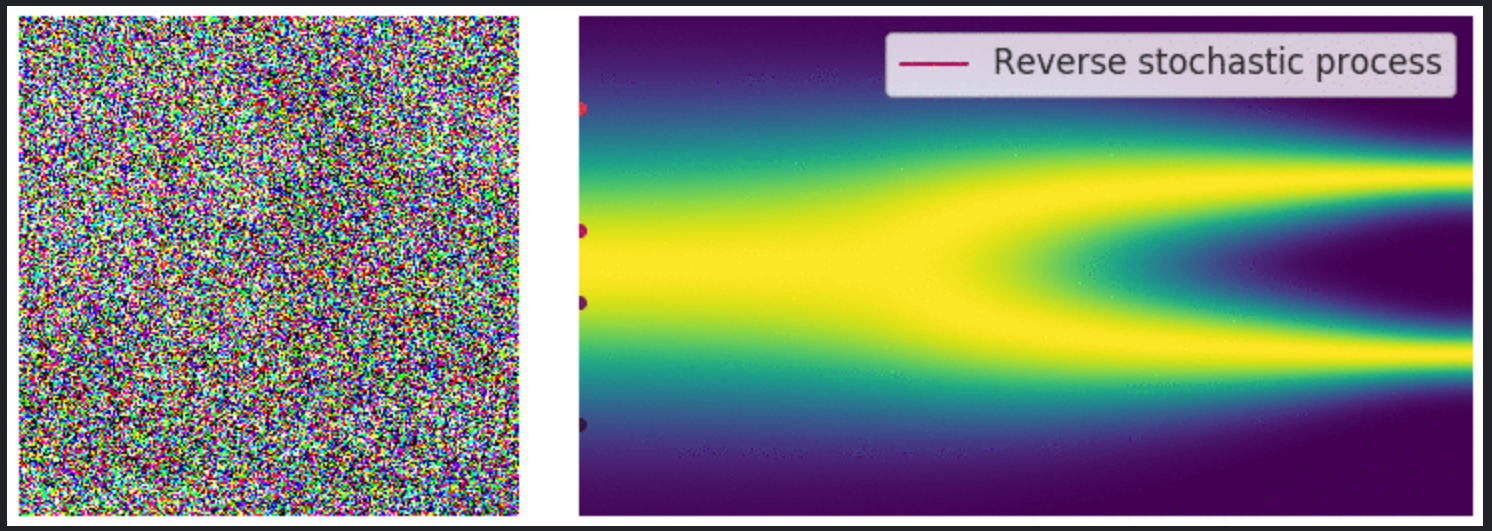
Mathematically, the forward process can be expressed as an SDE, e.g.

$$dy(t) = dW_t$$
$$y(0) \sim q$$

We can reverse this SDE to obtain another SDE, or an ODE

$$dy(t) = -\frac{1}{2}\nabla_y \log p_t(y) dt,$$





https://yang-song.net/blog/2021/score/#score-based-generative-modeling-with-stochastic-differential-equations-sdes

Score-based Diffusion Models

Approximate the *score* $\nabla_y \log p_t(y)$ by a neural network $s(y, t)$

(This is a Neural ODE!)

Controlled Generation
Use conditional score

$$\nabla_y \log p_t(y|x) = \nabla_y \log p_t(y) + \nabla_y \log p_t(x|y)$$

Closing words

'The Euler-Lagrange Equations, oil painting, trending on artstation, HQ", made with Stable Diffusion

Further Reading

- P. Kidger, *On Neural Differential Equations*, PhD Thesis (2022)
- Rackauckas et al. *Universal Differential Equations for Scientific Machine Learning*, (2020)
- P. Kidger - Score based diffusions explained in just one paragraph
- A. Ananthaswamy - The Physics Principle That inspired Modern AI Art - Quanta Magazine
- Cheatsheet: Recurrent Neural Networks