Neural Stochastic Differential Equations

Alexandra Volokhova

Bayesian methods research group
Moscow Institute of Physics and Technology
Yandex School of Data Analysis

p(B|A)yesgroup.ru

MIPT

Yandex School of Data Analysis

What is neural ODE?

usual neural net

$$f(input) = output$$

$$input \longrightarrow \boxed{f} \longrightarrow output$$

What is neural ODE?

usual neural net

$$f(input) = output$$

$$input \longrightarrow \boxed{f} \longrightarrow output$$

neural ODE

$$\frac{dz}{dt} = f(z(t), t)$$

$$z(t),\ t \longrightarrow \boxed{f} \longrightarrow \frac{dz}{dt}(t)$$

$$z(0) = input$$

$$output = z(T)$$

What is neural ODE?

usual neural net

$$f(input) = output$$

$$input \longrightarrow \boxed{f} \longrightarrow output$$

neural ODE

$$\frac{dz}{dt} = f(z(t), t)$$

$$z(t),\ t \longrightarrow \boxed{f} \longrightarrow \frac{dz}{dt}(t)$$

$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f(z(t), t)dt$$

What is neural ODE?

usual neural net

$$f(input) = output$$

$$input \longrightarrow \boxed{f} \longrightarrow output$$

neural ODE

$$\frac{dz}{dt} = f(z(t), t)$$

$$z(t),\ t \longrightarrow \boxed{f} \longrightarrow \frac{dz}{dt}(t)$$

$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f(z(t), t)dt$$

$$\boxed{output = ODESolver(z(0), f, T)}$$

Pros and cons of neural ODE

- precise integration is **very** slow
- not outperform usual neural nets

neural ODE

$$\frac{dz}{dt} = f(z(t), t)$$

$$z(t),\ t \longrightarrow \boxed{f} \longrightarrow \frac{dz}{dt}(t)$$

$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f(z(t), t)dt$$

$$\boxed{output = ODESolver(z(0), f, T)}$$

Pros and cons of neural ODE

- precise integration is **very** slow
- not outperform usual neural nets

✓ save memory during the training (adjoint method)

✓ work with time series quite well

neural ODE

$$\frac{dz}{dt} = f(z(t), t)$$

$$z(t),\ t \longrightarrow \boxed{f} \longrightarrow \frac{dz}{dt}(t)$$

$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f(z(t), t)dt$$

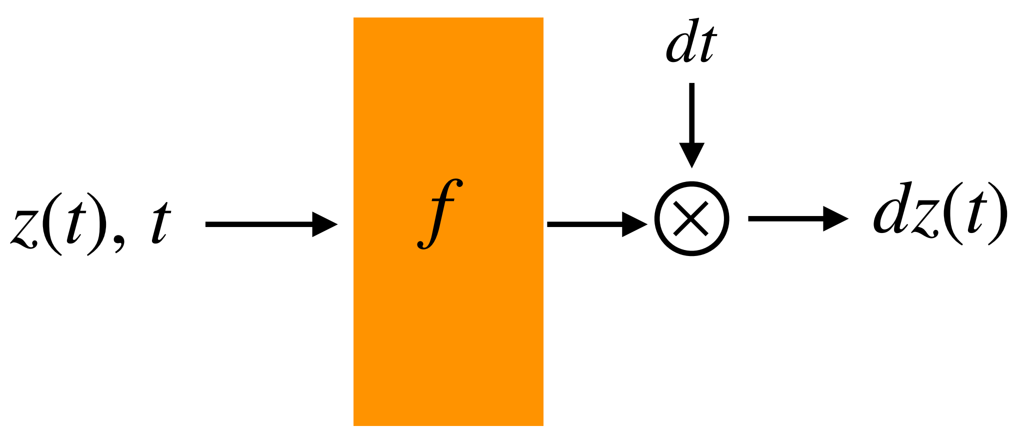
$$\boxed{output = ODESolver(z(0), f, T)}$$

Neural Stochastic Differential Equations

What is neural SDE?

neural ODE

$$dz = f\big(z(t), t\big)dt$$



$$z(0) = input$$

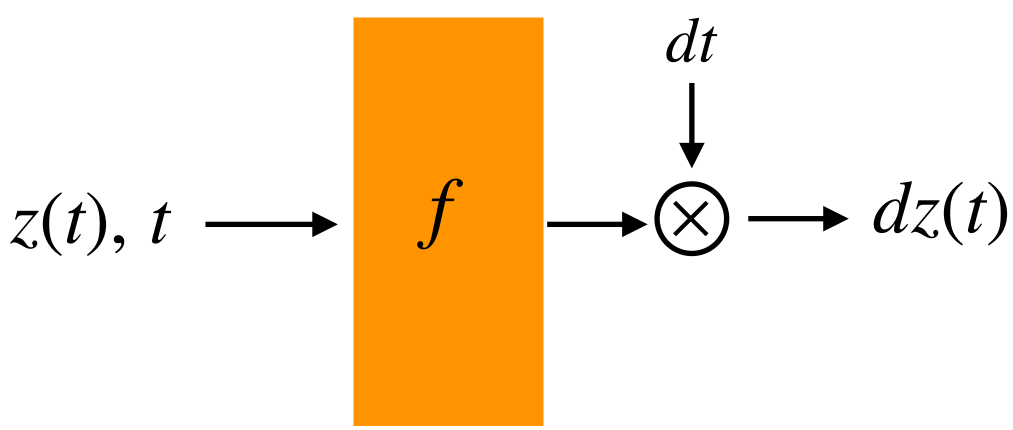
$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f\big(z(t), t\big)dt$$

What is neural SDE?

neural ODE

$$dz = f\big(z(t), t\big)dt$$



$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f\big(z(t), t\big)dt$$

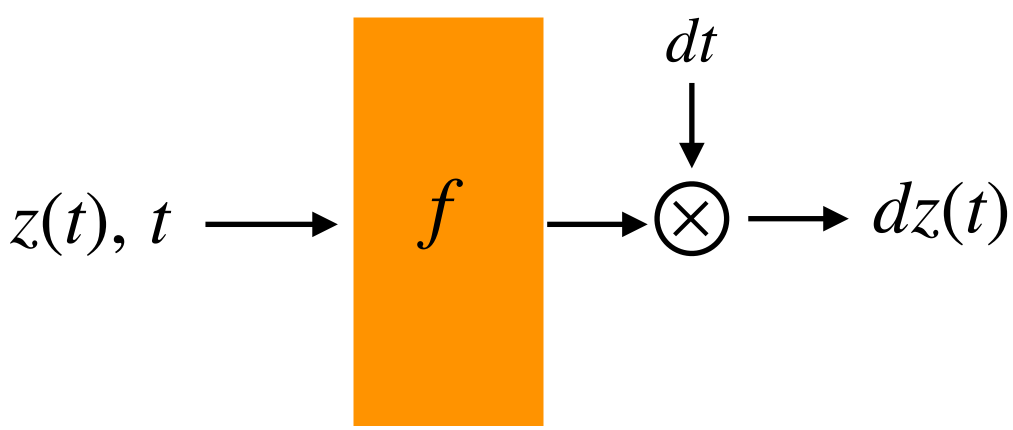
neural SDE

$$dz = f\big(z(t), t\big)dt + \sigma\big(z(t), t\big) \odot dW$$

What is neural SDE?

neural ODE

$$dz = f\big(z(t), t\big)dt$$



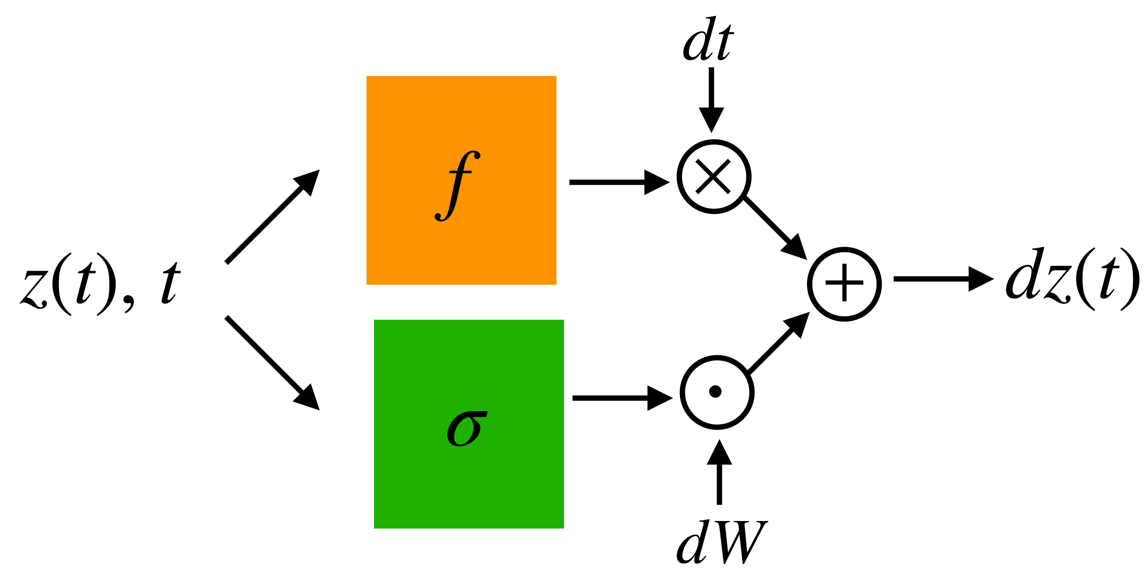
$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f\big(z(t), t\big)dt$$

neural SDE

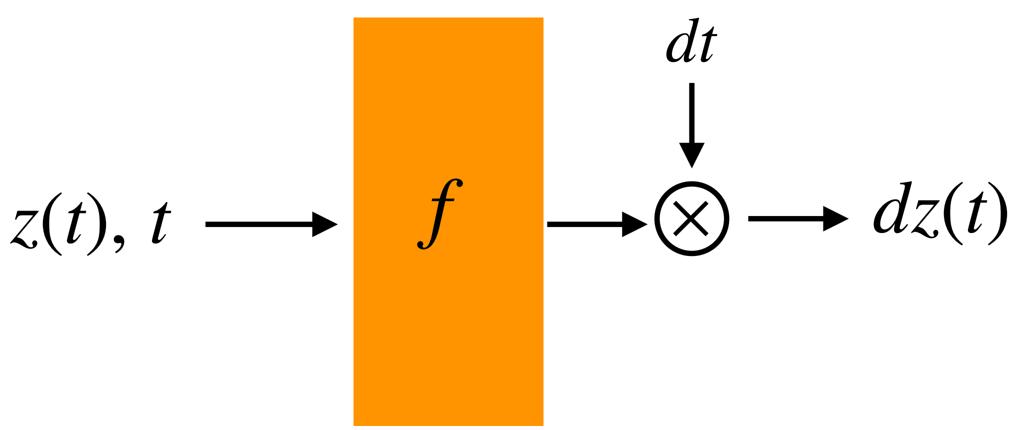
$$dz = f\big(z(t), t\big)dt + \sigma\big(z(t), t\big) \odot dW$$



What is neural SDE?

neural ODE

$$dz = f\big(z(t), t\big)dt$$



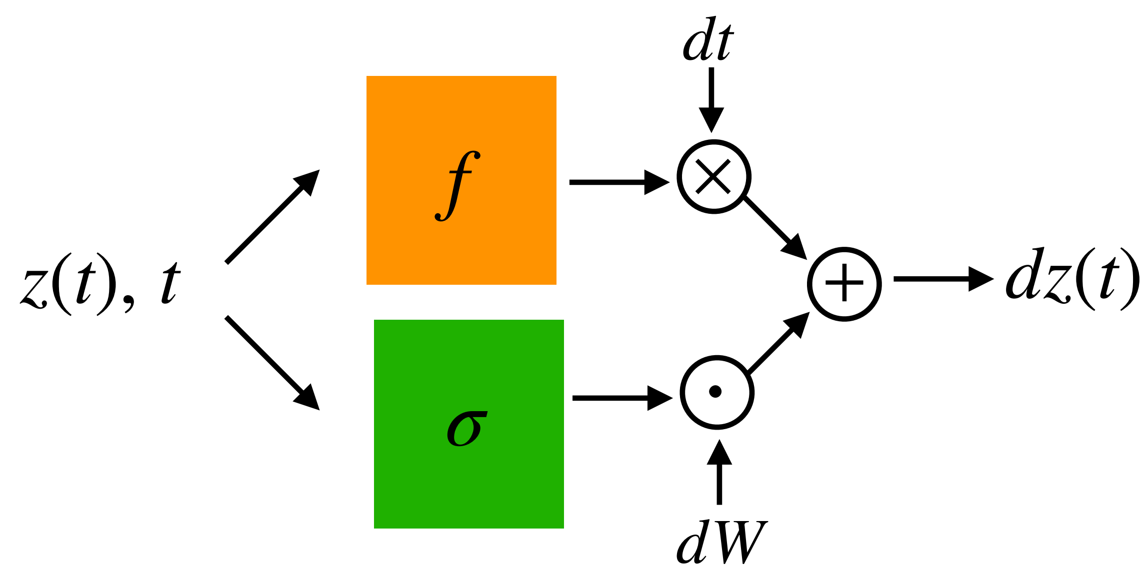
$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f\big(z(t), t\big)dt$$

neural SDE

$$dz = f\big(z(t), t\big)dt + \sigma\big(z(t), t\big) \odot dW$$

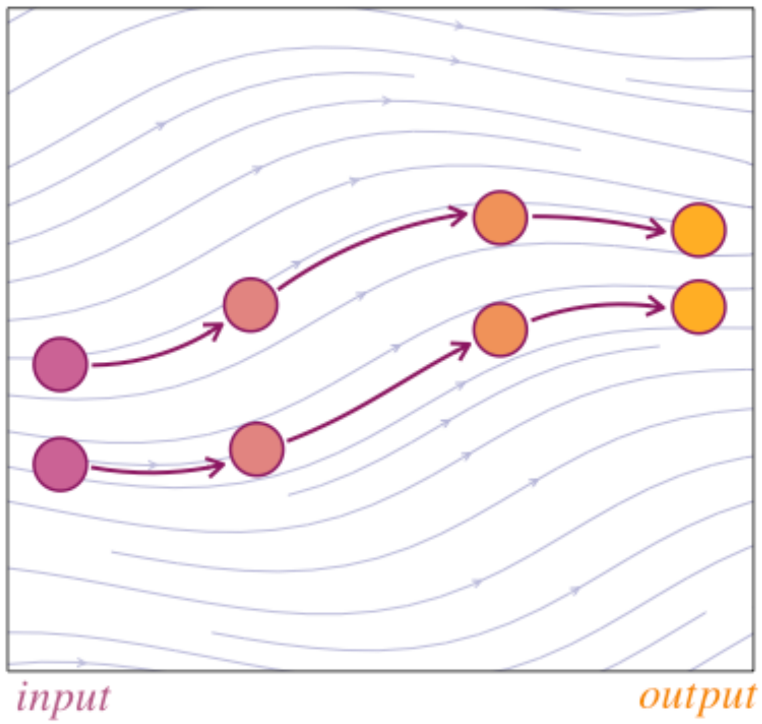


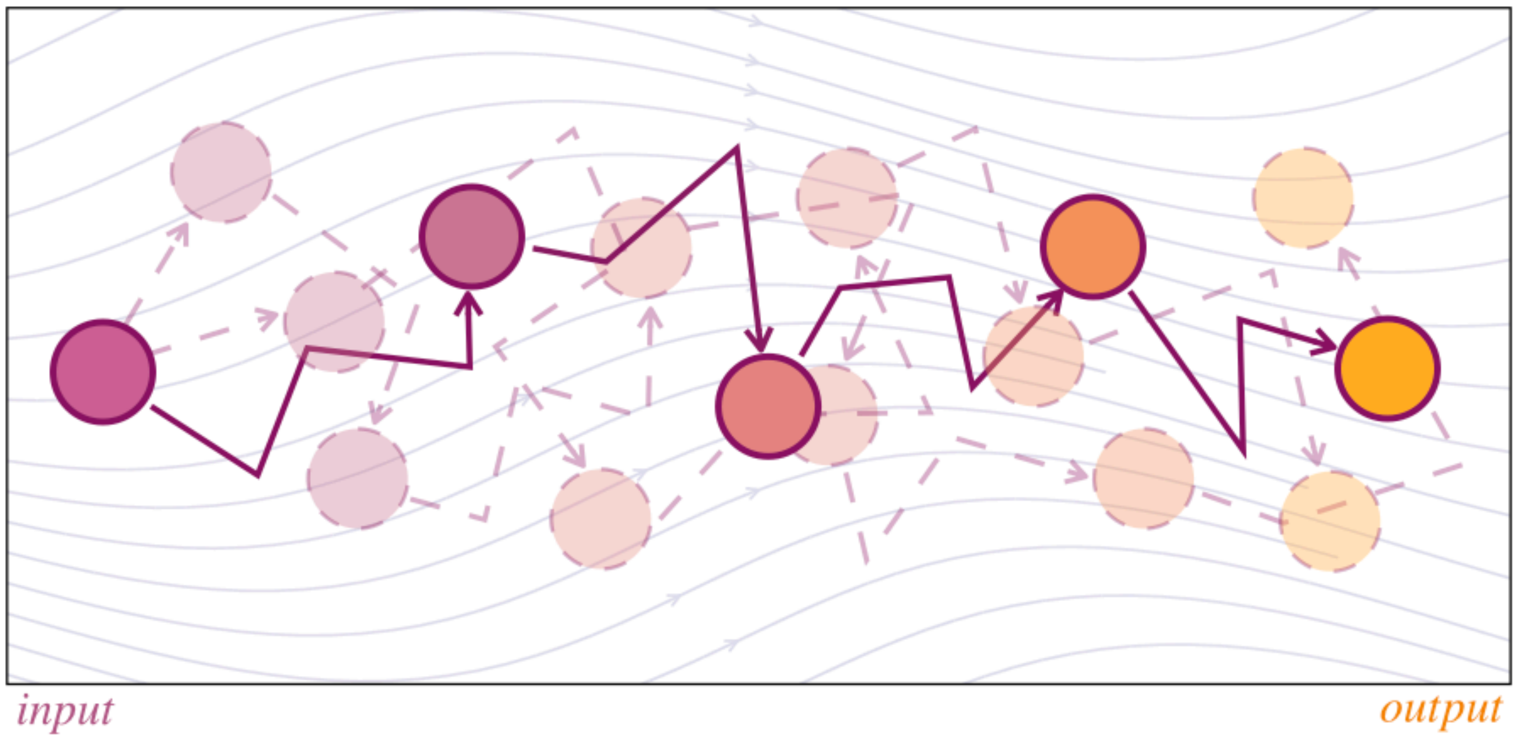
$$z(0) = input$$

$$output = z(T)$$

$$z(T) = z(0) + \int_0^T f\big(z(t), t\big)dt + \int_0^T \sigma\big(z(t), t\big) \odot dW$$

What is neural SDE?





neural ODE

$$dz = f\big(z(t), t\big)dt$$

neural SDE

$$dz = f\big(z(t), t\big)dt + \sigma\big(z(t), t\big) \odot dW$$

✓ possible regularization properties

✓ averaging along random trajectories

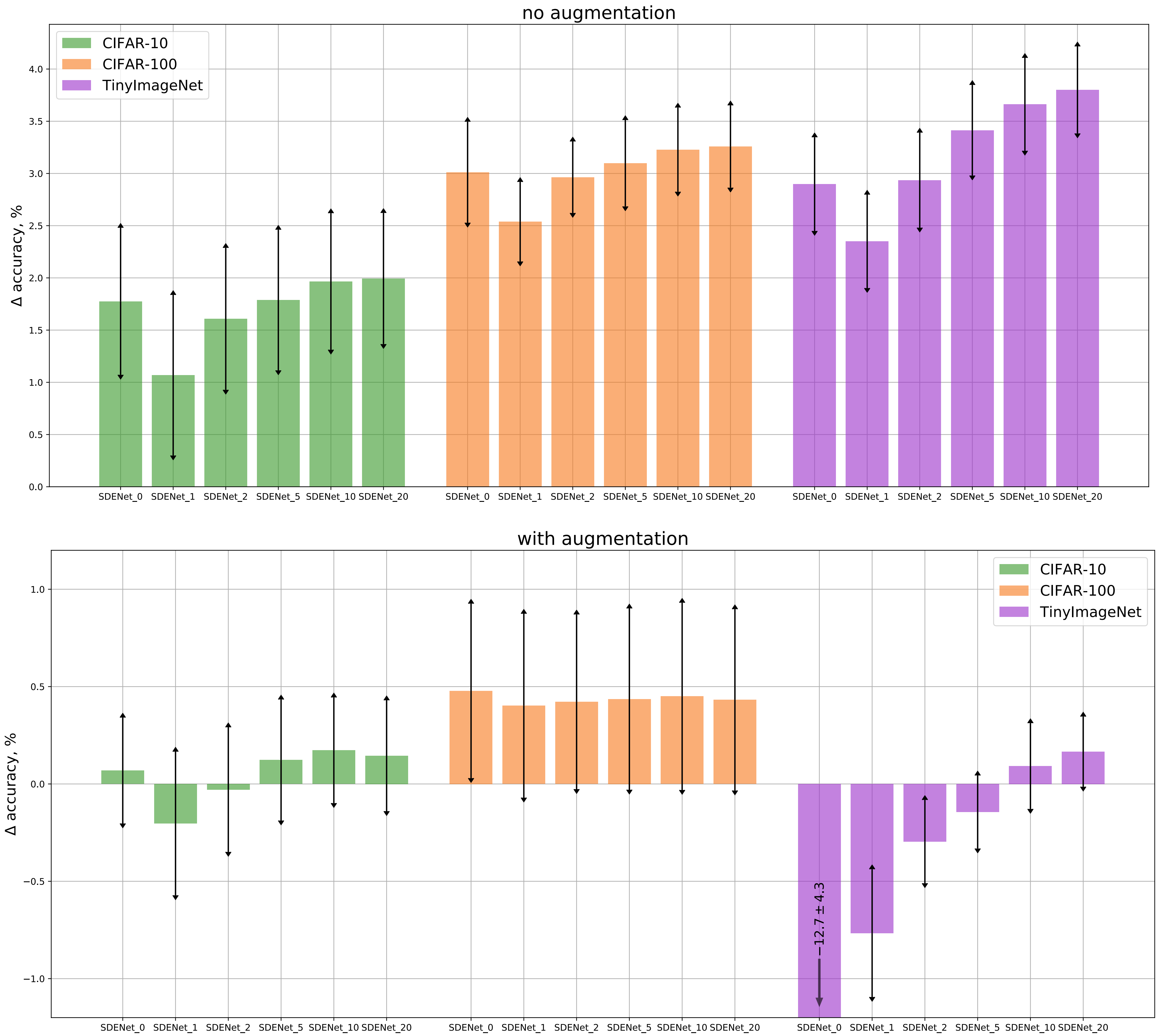
Regularization properties of neural SDE

$$dz = f\big(z(t), t\big)dt + \sigma dW$$

	CIFAR-10		CIFAR-100		TinyImageNet	
	augment	no augment	augment	no augment	augment	no augment
ResNet	90.6 ± 0.2	85.5 ± 0.5	76.5 ± 0.4	63.8 ± 0.5	50.4 ± 0.3	39.5 ± 0.4
ODENet	90.9 ± 0.2	85.0 ± 0.6	74.7 ± 0.4	62.5 ± 0.3	50.1 ± 0.1	36.8 ± 0.4
ODENet+BN	72.6 ± 4.3	48.0 ± 10.0	70.2 ± 1.4	55.7 ± 0.9	50.6 ± 0.4	36.7 ± 2.7
SDENet_0	90.9 ± 0.1	86.7 ± 0.4	75.2 ± 0.1	65.5 ± 0.4	37.4 ± 4.3	39.7 ± 0.2
SDENet_1	90.7 ± 0.3	86.0 ± 0.5	75.1 ± 0.2	65.0 ± 0.3	49.4 ± 0.3	39.1 ± 0.2
SDENet_2	90.8 ± 0.2	86.6 ± 0.4	75.2 ± 0.1	65.4 ± 0.2	49.8 ± 0.2	39.7 ± 0.2
SDENet_5	91.0 ± 0.2	86.7 ± 0.4	75.2 ± 0.2	65.6 ± 0.3	50.0 ± 0.2	40.2 ± 0.2
SDENet_10	91.0 ± 0.1	86.9 ± 0.3	75.2 ± 0.2	65.7 ± 0.3	50.2 ± 0.2	40.4 ± 0.2
SDENet_20	91.0 ± 0.2	87.0 ± 0.3	75.2 ± 0.2	65.7 ± 0.3	50.3 ± 0.2	40.6 ± 0.1

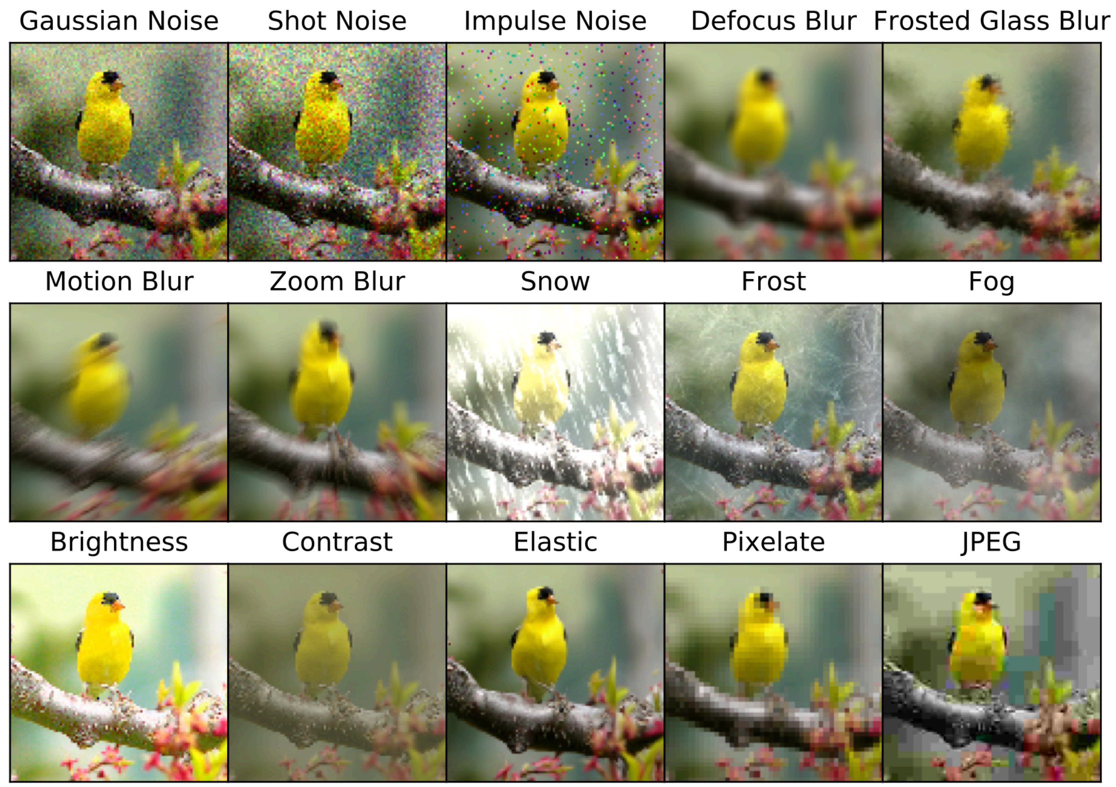
«Stochasticity in Neural ODEs: An Empirical Study» V.Oganesyan, A.Volokhova, D.Vetrov https://arxiv.org/abs/2002.09779

Regularization properties of neural SDE



Oganesyan V, Volokhova A, Vetrov D. Stochasticity in Neural ODEs: An Empirical Study. 2020 Feb 22. https://arxiv.org/abs/2002.09779

Robustness of neural SDE



https://github.com/hendrycks/robustness

Table 2: Testing accuracy results under different levels of non-adversarial perturbations.

Data	Noise type	mild corrupt ← Accuracy → severe corrupt				
		Level 1	Level 2	Level 3	Level 4	Level 5
CIFAR10-C†	ODE	75.89	70.59	66.52	60.91	53.02
	Dropout	77.02	71.58	67.21	61.61	53.81
	Dropout+TTN	**79.07**	**73.98**	**69.74**	**64.19**	**55.99**
TinyImageNet-C†	ODE	23.01	19.18	15.20	**12.20**	**9.88**
	Dropout	22.85	18.94	14.64	11.54	9.09
	Dropout+TTN	**23.84**	**19.89**	**15.28**	12.08	9.44

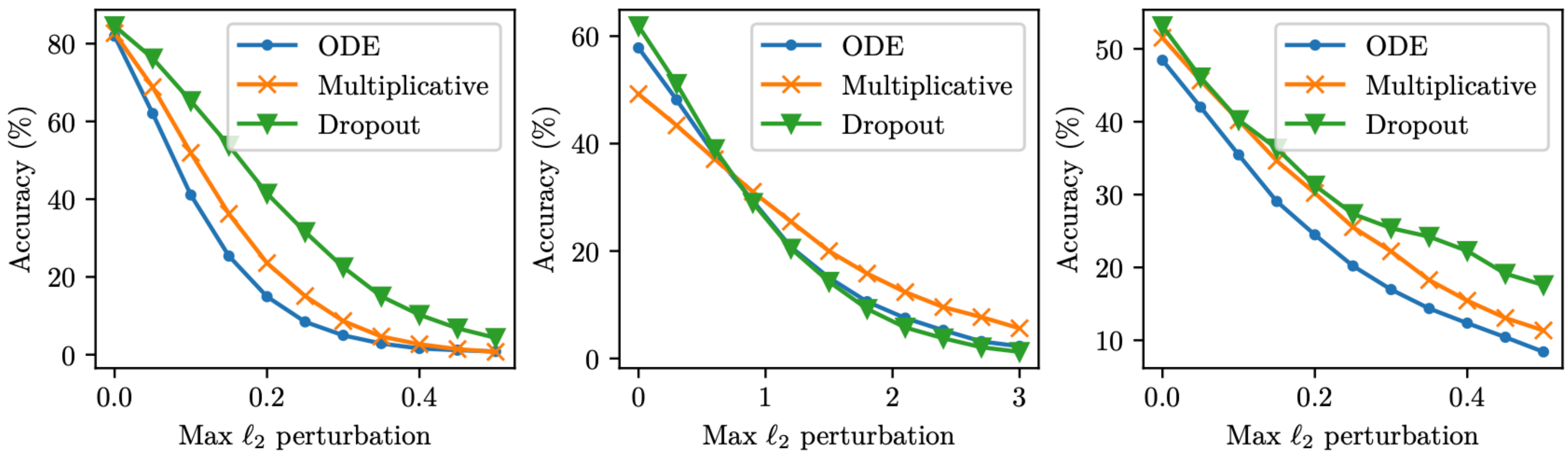
† Downloaded from `https://github.com/hendrycks/robustness`

Dropout:

$$dz = f\big(z(t), t\big)\,dt + \sqrt{\frac{1-p}{p}}\,f\big(z(t), t\big) \odot dW$$

Liu, Xuanqing, et al. Neural SDE: Stabilizing Neural ODE Networks with Stochastic Noise. 2019 Jun 5 https://arxiv.org/abs/1906.02355

Robustness of neural SDE



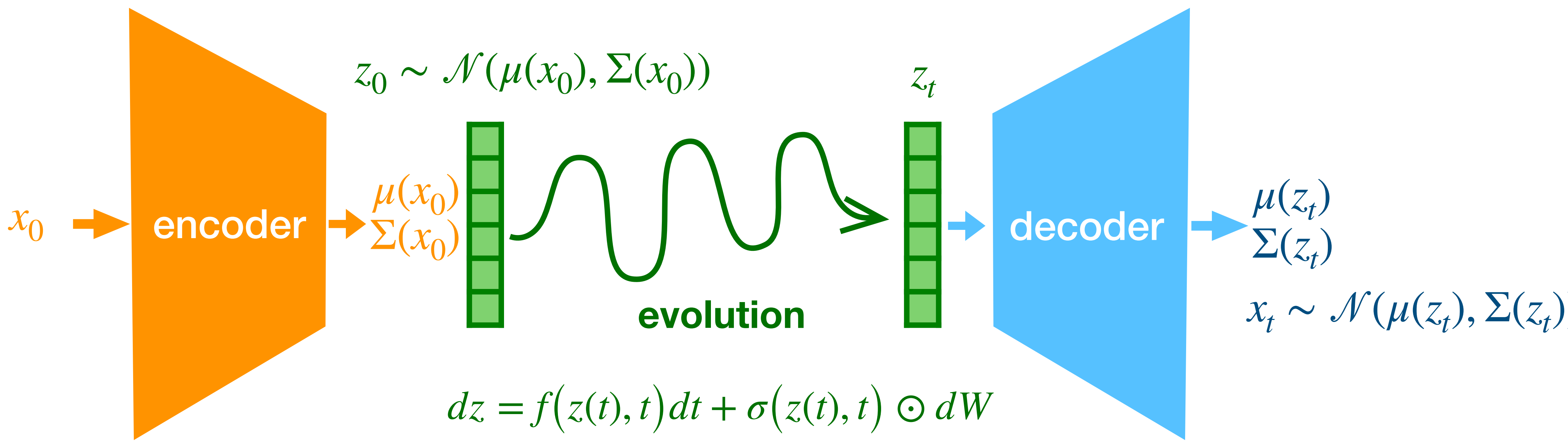
CIFAR-10 STL-10 TinyImageNet

Multiplicative: $$dz = f\big(z(t), t\big)dt + \sigma z(t) \odot dW$$

Dropout: $$dz = f\big(z(t), t\big)dt + \sqrt{\frac{1-p}{p}} f\big(z(t), t\big) \odot dW$$

Liu, Xuanqing, et al. Neural SDE: Stabilizing Neural ODE Networks with Stochastic Noise. 2019 Jun 5 https://arxiv.org/abs/1906.02355

Neural SDE for time-series



$$ELBO = \frac{1}{T+1} \sum_{t=0}^{T} \left[\mathbb{E}_{q(z_0|x_0)} \log p(x_t \,|\, z_t) \right] - KL(q(z_0) \,||\, p(z_0))$$

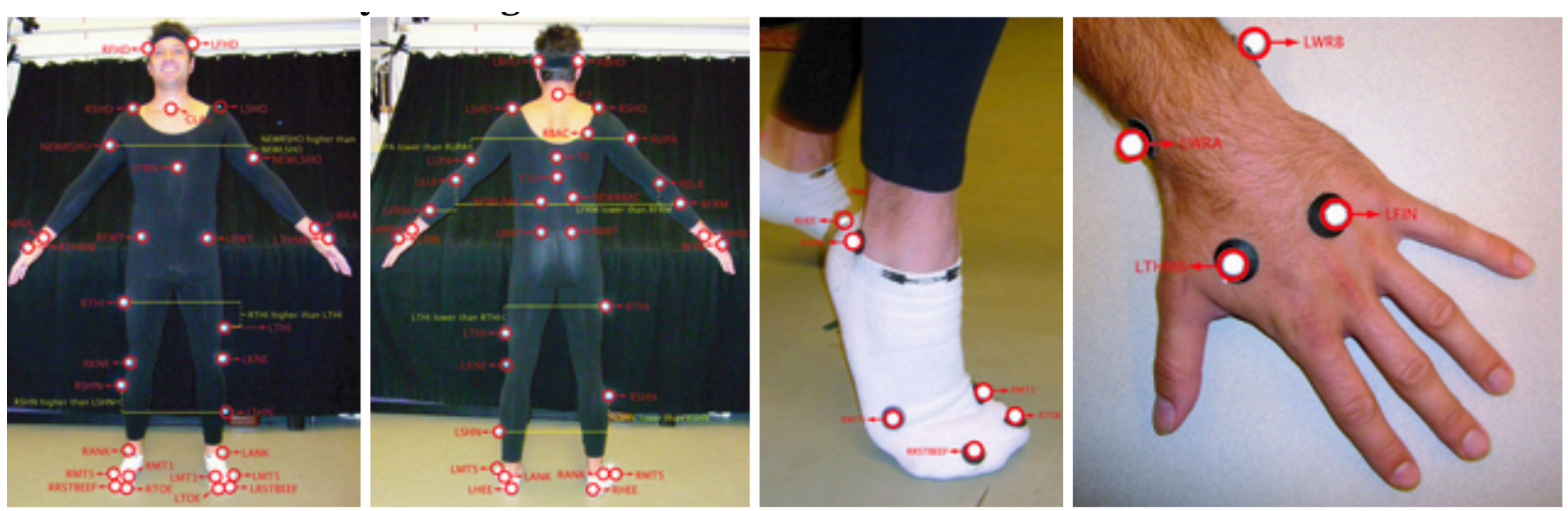
Li, Xuechen, et al. "Scalable Gradients for Stochastic Differential Equations." *arXiv preprint arXiv:2001.01328* (2020)

Neural SDE for time-series

Table 2: Test MSE on 297 future frames averaged over 50 samples. 95% confidence interval reported based on t-statistic. †results from [86].

Method	Test MSE
DTSBN-S [17]	$34.86 \pm 0.02^{\dagger}$
npODE [27]	$22.96^{\dagger}$
NeuralODE [11]	$22.49 \pm 0.88^{\dagger}$
ODE^{2}VAE [86]	$10.06 \pm 1.4^{\dagger}$
ODE^{2}VAE-KL [86]	$8.09 \pm 1.95^{\dagger}$
Latent ODE [11, 69]	5.98 ± 0.28
Latent SDE (this work)	$\mathbf{4.03 \pm 0.20}$

Performance at CMU Motion capture dataset



Li, Xuechen, et al. "Scalable Gradients for Stochastic Differential Equations." *arXiv preprint arXiv:2001.01328* (2020)

Summary

✓ Neural SDE performs better than Neural ODE **in some cases** (without augmentation)

✓ Neural SDE works with time-series quite well