Deep Learning for Computing Convergence Rates of Markov Chains

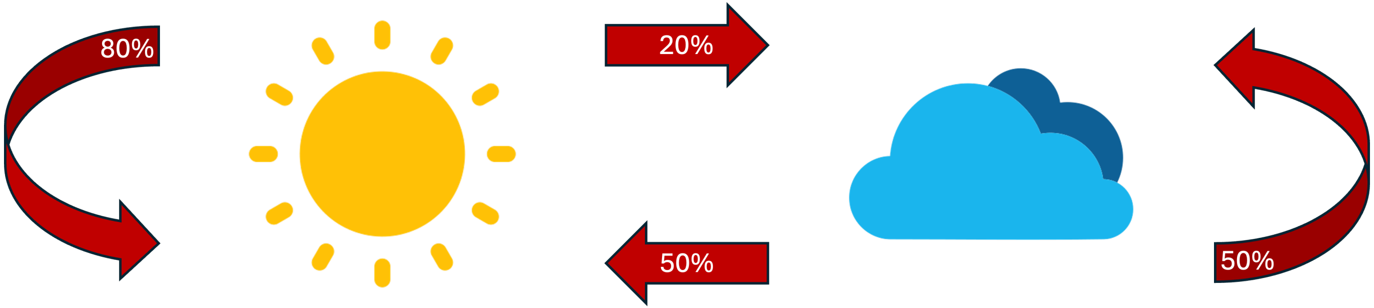
Yanlin Qu, Jose Blanchet, Peter Glynn

Stanford University

NeurIPS 2024 Spotlight

Stanford ENGINEERING
Management Science & Engineering

Example of a Markov Chain



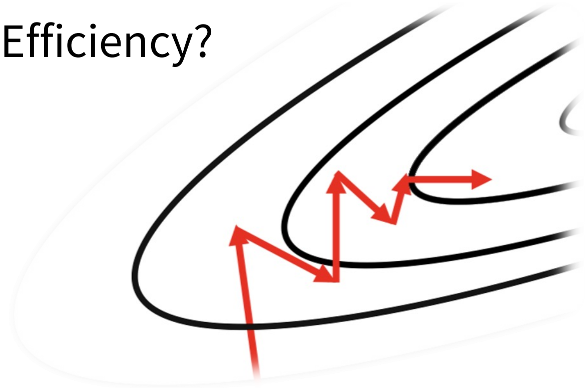
What is the percentage of sunny days in the long run?

$$P(X_n = \text{☀}) \rightarrow P(X_\infty = \text{☀}) = 5/7$$

Importance of Convergence Analysis

Reliability?

Efficiency?

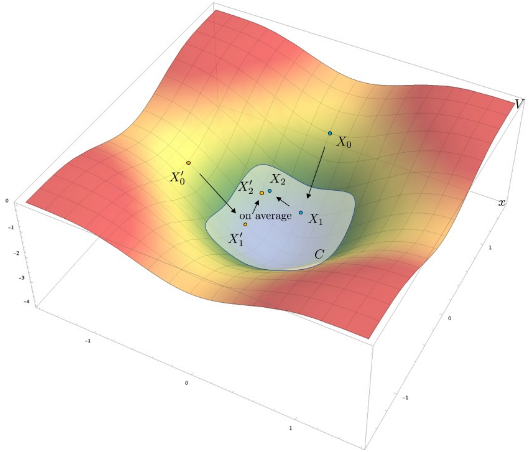


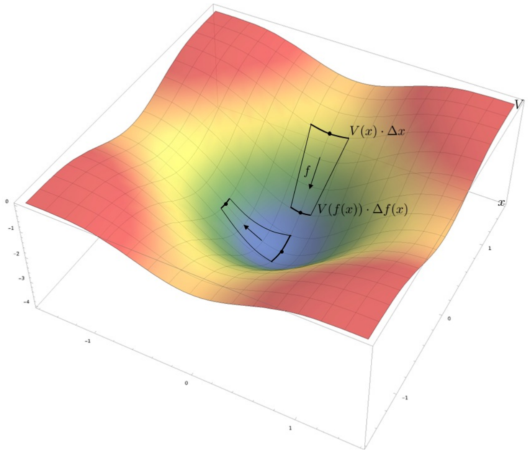
$$X_{n+1} = (X_n + S_{n+1} - A_{n+1})_+$$

$$X_{n+1} = X_n - \alpha \nabla L(X_n, Z_{n+1})$$

$$\boxed{\textit{How fast does } X_n \textit{ converge to } X_\infty\textit{?}}$$

Convergence Analysis: Old and New





Drift & Contraction, Hairer et al. (2011)

$$PV(x) = \mathbb{E}_x V(X_1) \leq V(x) - U(x), \quad x \notin C$$
$$W(P(y,\cdot), P(z,\cdot)) \leq \alpha d(y,z), \quad y, z \in C$$

Contractive Drift (CD), Qu et al. (2023)

$$KV(x) = \mathbb{E}_x Df(x) V(f(x)) \leq V(x) - U(x)$$
$$Df(x) \approx \Delta f(x)/\Delta x, \quad X_{n+1} = f_{n+1}(X_n)$$

Hairer, M., Mattingly, J. C. & Scheutzow, M. (2011), Asymptotic coupling and a general form of Harris' theorem with applications to stochastic delay equations, Probability Theory and Related Fields 149(1), 223–259.
Qu, Y., Blanchet, J. & Glynn, P. (2023), Computable bounds on convergence of Markov chains in Wasserstein distance, arXiv:2308.10341.

Stanford | ENGINEERING
Management Science & Engineering

From Pen and Paper to Deep Learning

Q, Blanchet, Glynn (2024)

Stanford | ENGINEERING
Management Science & Engineering

Contractive Drift Equation (CDE)

- Let $X_{n+1} = f_{n+1}(X_n)$ be a Markov chain on $\mathcal{X} \subset \mathbb{R}^d$.
- If f is differentiable, then $Df(x) = \|\nabla f(x)\|$.
- CD is actually CDE: $KV(x) = \mathbb{E}Df(x)V(f(x)) = V(x) - U(x)$.

Theorem. *Fix U and suppose that $KW \leq W - U$ has a non-negative finite solution W_*. Then*

$$V_*(x) \triangleq \mathbb{E}_x \left[\sum_{k=0}^{\infty} U(X_k) \prod_{l=1}^{k} Df_l(X_{l-1}) \right], \quad x \in \mathcal{X}$$

is finite and satisfies $KV_* = V_* - U$. Furthermore, $KV = V - U$ has at most one bounded solution.

Why do we introduce CDE?

- Physics-informed neural networks (PINNs) solve a PDE by minimizing its integrated $\boxed{\text{squared}}$ residual; see Raissi et al. (2019).

- The residual of a CDE is ($X_0 \sim h$)

$$
\begin{aligned}
2l(\theta) =& \int_{\mathcal{X}} (KV_\theta(x) - V_\theta(x) + U(x))^2 h(x) dx \\
=& \mathbb{E}\left[\mathbb{E}\left[Df_1(X_0)V_\theta(f_1(X_0)) - V_\theta(X_0) + U(X_0)|X_0\right]\right]^2 .
\end{aligned}
$$

- Its derivative $l'(\theta)$ is (f_1, f_{-1} iid)

$$
\mathbb{E}\left[\left[Df_1(X_0)V_\theta(f_1(X_0)) - V_\theta(X_0) + U(X_0)\right]\left[Df_{-1}(X_0)V'_\theta(f_{-1}(X_0)) - V'_\theta(X_0)\right]\right],
$$

which leads to an $\boxed{\text{unbiased}}$ gradient estimator.

Raissi, M., Perdikaris, P. & Karniadakis, G. E. (2019), Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, Journal of Computational Physics 378, 686–707.

Deep Contractive Drift Calculator (DCDC)

Require: Step-size α, number of iterations T, neural network $\{V_\theta : \theta \in \Theta\}$, initialization θ_0

for $t \in \{0, ..., T-1\}$ **do**

sample (X_0, f_1, f_{-1})

compute $\hat{l}'(\theta_t)$ as

$$[Df_1(X_0)V_{\theta_t}(f_1(X_0)) - V_{\theta_t}(X_0) + U(X_0)]\,[Df_{-1}(X_0)V'_{\theta_t}(f_{-1}(X_0)) - V'_{\theta_t}(X_0)]$$

update $\theta_{t+1} = \theta_t - \alpha\hat{l}'(\theta_t)$ (SGD or its variants)

end for

convert V_{θ_T} into a convergence bound

$$W(X_n, X_\infty) \leq Cr^n, \quad r = 1 - \inf U / \sup V, \quad C = \frac{\mathbb{E}\,\|X_0 - X_1\|\, V(X_0 + \tilde{U}(X_1 - X_0))}{\inf U \cdot (\inf V / \sup V)}.$$

Stanford | ENGINEERING
Management Science & Engineering

A Realistic SGD Example

- Data: $(x_1, y_1), \ldots, (x_m, y_m) \in [-1/2, 1/2]^2 \times \{0, 1\}$
- Regularized logistic loss:

$$-\frac{1}{m}\sum_{i=1}^{m}(y_i \log p_i + (1 - y_i)\log(1 - p_i)) + \frac{\lambda}{2m}\|b\|^2, \quad p_i = \sigma(b^\top x_i)$$

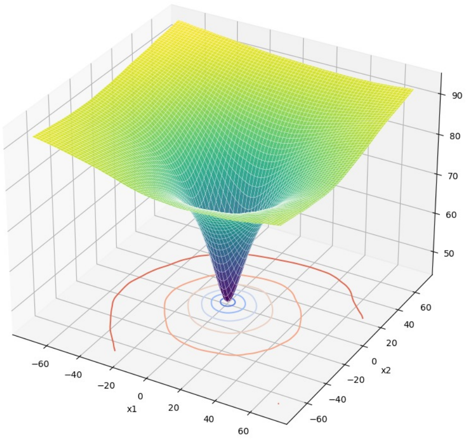
- SGD with step-size α and batch-size β:

$$f(b) = b(1 - \lambda\alpha/m) + (\alpha/\beta)\sum_{i \in B}\left[y_i - \sigma(b^\top x_i)\right] x_i$$

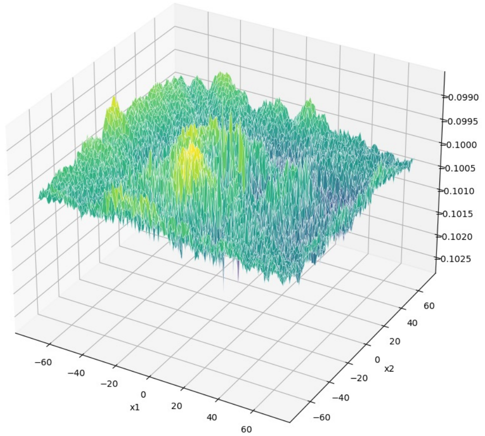
- $m = 100,\ \lambda = 1,\ \alpha = 0.1,\ \beta = 10$

A Realistic SGD Example

- A single-layer network with width 1000 and sigmoid activation
- $W(X_n, X_\infty) \leq 8.1(1 - 1.07 \times 10^{-3})^n$



Learned solution of $KV = V - 0.1$



Estimated difference $\hat{K}\tilde{V} - \tilde{V}$

Stanford ENGINEERING
Management Science & Engineering

Takeaway

DCDC is just a start, the start of **computational** Markov chain convergence analysis.

Thank you

https://quyanlin.github.io/

Stanford | ENGINEERING
Management Science & Engineering