Stochastic Optimisation

Ishan Kapnadak

Spring Semester 2021-22

Updated on: 2022-02-18

Abstract

Lecture Notes for the course EE 736 : Stochastic Optimisation taught in Spring 2022 by Prof. Vivek Borkar.

Contents

Chapter 1

Stochastic Approximation

§1.1. The Robbins-Monro Algorithm

The basic problem we consider is to solve $h(\mathbf{x}) = 0$ given noisy measurements of h. That is, we are given access to a black box that, on input $\mathbf{x} \in \mathbb{R}^d$, gives as output $h(\mathbf{x}) + \text{noise}$. To this end, we have the *Robbins-Monro algorithm*.

Robbins-Monro Algorithm. Starting with $\mathbf{x}_0 \in \mathbb{R}^d$, do:

$$\mathbf{x}(n+1) := \mathbf{x}(n) + a(n)\left[h(\mathbf{x}(n)) + M(n+1)\right], \quad n \geq 0.$$

Here, the (non-negative) stepsize sequence (or learning parameter) $\{a(n)\}$ satisfies

$$\sum_n a(n) = \infty \quad \text{and} \quad \sum_n a(n)^2 < \infty.$$

A typical example of such a stepsize sequence is $\frac{1}{n}, \frac{1}{n \log n}, \frac{1}{n^{2/3}}$, and so on. Further, we make the following assumptions.

1. $h\colon \mathbb{R}^d \to \mathbb{R}^d$ is Lipschitz, that is, $\exists L \geq 0$ such that

   $$\|h(\mathbf{x}) - h(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$$

   for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

2. $\{M(n)\}$ is a square integrable martingale difference sequence. That is, for

   $$\mathcal{F}_n := \sigma\left(\mathbf{x}_0, M_m, m \leq n\right), n \geq 0$$

   we have

   $$\mathbb{E}\left[\|M(n)\|^2\right] < \infty$$

and in addition, we have that it is uncorrelated with the past. That is,

$$\mathbb{E}\left[M_i(n+1) \mid \mathcal{F}_n\right] = 0 \quad \forall i.$$

Furthermore, we assume that for some $K > 0$,

$$\mathbb{E}\left[\|M(n+1)\|^2 \mid \mathcal{F}_n\right] \leq K(1 + \|\mathbf{x}(n)\|^2) \quad \forall n \geq 0.$$

In particular, if

$$\sup_n \|\mathbf{x}(n)\| < \infty \quad \text{a.s.},$$

then,

$$\sup_n \mathbb{E}\left[\|M(n+1)\|^2 \mid \mathcal{F}_n\right] < \infty \quad \text{a.s..}$$

This algorithm is actually more general than it appears. Suppose the algorithm is

$$\mathbf{x}(n+1) := \mathbf{x}(n) + a(n) f\left(\mathbf{x}(n), \xi(n+1)\right), \quad n \geq 0$$

where $\{\xi(n)\}$ are i.i.d. This is often how most recursive algorithms are stated. The above algorithm can be put into the form of Robbins-Monro algorithm by choosing

$$\begin{aligned} h(\mathbf{x}) &:= \mathbb{E}\left[f(\mathbf{x}, \xi(n))\right] \\ &= \mathbb{E}\left[f(\mathbf{x}(n), \xi(n+1)) \mid \mathbf{x}(n) = \mathbf{x}\right] \\ &= \mathbb{E}\left[f(\mathbf{x}(n), \xi(n+1)) \mid \mathcal{F}_n\right] \end{aligned}$$

and

$$M(n+1) := f(\mathbf{x}(n), \xi(n+1)) - h(\mathbf{x}(n)).$$

A common example of Robbins-Monro algorithm is stochastic gradient descent, where we set $h = -\nabla f$. Robbins-Monro algorithm also finds uses in many reinforcement learning algorithms. Some advantages of the Robbins-Monro algorithm are listed as follows.

1. It typically requires a small amount of computation and memory per iterate.
2. It is incremental in nature, that is, it makes only a small change in the current iterate at each step.
3. The slowly decreasing stepsize $\{a(n)\}$ captures the exploration-exploitation trade-off.
4. It averages out the noise, which can be thought of as a generalisation of the Strong Law of Large Numbers.

Another common approach to solving the same problem is the ODE (Ordinary Differential Equation) approach, which treats the iterate as a noisy discretisation of the ODE

$$\dot{\mathbf{x}}(t) = h(\mathbf{x}(t)).$$

Recall the Euler scheme for solving this ODE:

$$\mathbf{x}(n+1) := \mathbf{x}(n) + ah(\mathbf{x}(n)), \quad n \geq 0,$$

where $a > 0$ is a small discrete time step. Thus the Robbins-Monro algorithm can be viewed as a Euler scheme to approximate the ODE with slowly decreasing time steps $\{a(n)\}$ and measurement noise. With this in mind, we have the following interpretation of the Robbins-Monro conditions on the step size $\{a(n)\}$.

1. $\sum_n a(n) = \infty$ ensures that the entire time axis is covered. This is essential because we want to track the asymptotic behaviour of the ODE.
2. $\sum_n a(n)^2 < \infty$ ensures that the approximation of the ODE gets better with time. In particular, $a(n) \to 0$ ensures that errors due to discretisation are asymptotically zero, and $\sum_n a(n)^2 < \infty$ ensures that errors due to the martingale difference noise are asymptotically zero almost surely, since multiplication by $a(n)$ reduces the conditional variance of the noise.

As an example, consider an initially empty urn to which one ball, either red or blue, is added at each time step. Let

$$\xi(n) := \mathbb{I}\{n^{\text{th}} \text{ ball is red}\} = \begin{cases} 1 & \text{if the } n^{\text{th}} \text{ ball is red, and} \\ 0 & \text{otherwise.} \end{cases}$$

Let $S(n) := \sum_{m=1}^{n} \xi(m)$ be the total number of red balls at time n, and let $x(n) := \frac{S(n)}{n}$ be the fraction of red balls at time n. Then, we have

$$\begin{aligned} x(n+1) &= \frac{1}{n+1} \sum_{m=1}^{n+1} \xi(m) \\ &= \frac{1}{n+1} \sum_{m=1}^{n} \xi(m) + \frac{\xi(n+1)}{n+1} \\ &= \left(\frac{n}{n+1}\right) \frac{\sum_{m=1}^{n} \xi(m)}{n} + \frac{\xi(n+1)}{n+1} \\ &= \left(1 - \frac{1}{n+1}\right) x(n) + \frac{\xi(n+1)}{n+1} \\ &= x(n) + a(n)(\xi(n+1) - x(n)) \end{aligned}$$

for $a(n) := \frac{1}{n+1}$ which satisfies the Robbins-Monro conditions. Now, suppose that

$$\mathbb{P}\left(\xi(n+1) = 1 \mid \xi(m), m \leq n\right) = p(x(n))$$

for some continuously differentiable function $p\colon [0,1] \to [0,1]$. Then, we have

$$\begin{aligned} x(n+1) &= x(n) + a(n)(\xi(n+1) - x(n)) \\ &= x(n) + a(n)\left[(p(x(n)) - x(n)) + (\xi(n+1) - p(x(n)))\right] \\ &= x(n) + a(n)\left[h(x(n)) + M(n+1)\right] \end{aligned}$$

for $h(x) := p(x) - x$, and $M(n+1) := \xi(n+1) - p(x(n))$. Since $\mathbb{E}\left[\xi(n+1) \mid \xi(m), m \leq n\right] = p(x(n))$ for all n, we have that $\{M(n)\}$ is a martingale difference sequence. Since $|M(n)| \leq 2$, the bound on conditional second moment is free. The limiting ODE is

$$\dot{x}(t) = p(x(t)) - x(t).$$

Under our hypothesis of continuous differentiability of p, this has a unique solution for any initial condition. Set $x(0) = x_0 \in [0,1]$. We have $p(0) - 0 \geq 0$, and $p(1) - 1 \leq 0$. Since $x(t) \in [0,1]$ for all $t \geq 0$, $x(t)$ must converge to a point in $[0,1]$. If at x_0, we have that $p(x_0) = x_0$, then we are already at equilibrium. If not, suppose that $p(x_0) > x_0$, then $x(t)$ is increasing but bounded by 1, so it must converge. A similar argument works for $p(x_0) < x_0$. But does an equilibrium exist? The answer is yes. Since $p(0) - 0 \geq 0$, and $p(1) - 1 \leq 0$, we have by continuity that there exists $x \in [0,1]$ such that $p(x) = x$. In fact, there can be more than one equilibria. An equilibrium x^* satisfies $p(x^*) = x^*$ and is stable if $p'(x^*) < 1$ and unstable if $p'(x^*) > 1$. Under some additional technicalities, we can show that $x(t)$ converges to one of the stable equilibria almost surely, and the probability of convergence to any stable equilibrium is strictly positive.

§1.2. Ordinary Differential Equations

We consider the ODE in $\mathbb{R}^d$, $d \geq 1$, given by

$$\dot{\mathbf{x}}(t) = h(\mathbf{x}(t)), \quad \mathbf{x}(0) = \mathbf{x}_0.$$

A problem is said to be *well-posed* if

1. it has a solution,
2. the solution is unique, and
3. the solution depends continuously on problem parameters.

For ODEs, this translates to the ODE having a unique solution for all time that depends continuously on the initial condition.

We say that $h\colon \mathbb{R}^d \to \mathbb{R}^d$ satisfies a (global) Lipschitz condition if for some $L > 0$, we have

$$\|h(\mathbf{x}) - h(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d.$$

h is locally Lipschitz if for all $R > 0$, there exists an $L_R > 0$ such that

$$\|h(\mathbf{x}) - h(\mathbf{y})\| \leq L_R\|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{B}_R := \left\{\mathbf{z} \in \mathbb{R}^d \mid \|\mathbf{z}\| \leq R\right\}$$

Lemma 1.2.1: Gronwall's Inequality

Suppose $0 \leq y\colon [0, T]\colon \mathbb{R}$ is differentiable and satisfies

$$y(t) \leq C + K\int_0^t y(s)\,\mathrm{d}s, \quad t \in [0, T]$$

for some $C, K > 0$. Then,

$$y(t) \leq Ce^{Kt}, \quad t \in [0, T].$$

Proof. Let $z(t) := \int_0^t y(s)\,\mathrm{d}s$, $t \geq 0$. Then,

$$\begin{aligned} \dot{z}(t) = y(t) &\leq C + Kz(t) \\ \implies e^{-Kt}(\dot{z}(t) - Kz(t)) &\leq Ce^{-Kt} \\ \implies \frac{\mathrm{d}}{\mathrm{d}t}\left(e^{-Kt}z(t)\right) &\leq Ce^{-Kt},\ z(0) = 0. \end{aligned}$$

Integrating both sides from 0 to t, we get

$$\begin{aligned} e^{-Kt}z(t) &\leq \frac{C}{K}(1 - e^{-Kt}) \\ \implies z(t) &\leq \frac{C}{K}(e^{Kt} - 1) \end{aligned}$$

Now, we have

$$\begin{aligned} y(t) \leq C + Kz(t) &\leq C + C(e^{Kt} - 1) \\ \implies y(t) &\leq Ce^{Kt}. \end{aligned}$$

□

Theorem 1.2.2

If h is Lipschitz, then the ODE $\{\dot{\mathbf{x}}(t) = h(\mathbf{x}(t)), \mathbf{x}(0) = \hat{\mathbf{x}}\}$ is well-posed.

Proof. We first show existence. Fix $T \in (0, 1/L)$ and a continuous function $\mathbf{x}_0 \colon [0,T] \to \mathbb{R}^d$ with $\mathbf{x}_0(0) = \hat{\mathbf{x}}$. Recursively define

$$\mathbf{x}_{n+1}(t) := \hat{\mathbf{x}} + \int_0^t h(\mathbf{x}_n(s))\,\mathrm{d}s, \quad t \in [0,T]. \tag{$\dagger$}$$

These are called *Picard iterations*. Then for $n \geq 1$, we have

$$\begin{aligned}
\|\mathbf{x}_{n+1}(t) - \mathbf{x}_n(t)\| &= \left\| \int_0^t (h(\mathbf{x}_n(s)) - h(\mathbf{x}_{n-1}(s)))\,\mathrm{d}s \right\| \\
&\leq \int_0^t \|h(\mathbf{x}_n(s)) - h(\mathbf{x}_{n-1}(s))\|\,\mathrm{d}s \\
&\leq L \int_0^t \|\mathbf{x}_n(s) - \mathbf{x}_{n-1}(s)\|\,\mathrm{d}s \\
&\leq LT \max_{s\in[0,T]} \|\mathbf{x}_n(s) - \mathbf{x}_{n-1}(s)\|
\end{aligned}$$

Thus,

$$\max_{t\in[0,T]} \|\mathbf{x}_{n+1}(t) - \mathbf{x}_n(t)\| \leq LT \max_{t\in[0,T]} \|\mathbf{x}_n(t) - \mathbf{x}_{n-1}(t)\|$$

Applying this repeatedly, we get

$$\begin{aligned}
\max_{t\in[0,T]} \|\mathbf{x}_{n+1}(t) - \mathbf{x}_n(t)\| \leq (LT)^n \max_{t\in[0,T]} \|\mathbf{x}_1(t) - \mathbf{x}_0(t)\| \text{ for } n \geq 0 \\
\implies \sum_{n=0}^{\infty} \max_{t\in[0,T]} \|\mathbf{x}_{n+1}(t) - \mathbf{x}_n(t)\| < \infty.
\end{aligned}$$

Thus, $\mathbf{x}_n(t) = \mathbf{x}_0(t) + \sum_{m=0}^{n-1}(\mathbf{x}_{m+1}(t) - \mathbf{x}_m(t))$ converges to some $\mathbf{x}(t)$ uniformly in $t \in [0,T]$. Passing to the limit as $n \uparrow \infty$ in ($\dagger$), we have

$$\mathbf{x}(t) := \hat{\mathbf{x}} + \int_0^t h(\mathbf{x}(s))\,\mathrm{d}s, \quad t \in [0,T]$$

Thus, $\mathbf{x}$ satisfies the ODE with $\mathbf{x}(0) = \hat{\mathbf{x}}$. We repeat the above procedure for $[T, 2T], [2T, 3T]$, and so on.

We now prove uniqueness. Consider $\dot{\mathbf{x}}(t) = h(\mathbf{x}(t))$, $\dot{\mathbf{y}}(t) = h(\mathbf{y}(t))$, $t \geq 0$ with $\mathbf{x}(0) = \mathbf{y}(0)$. Then,

$$\|\mathbf{x}(t) - \mathbf{y}(t)\| \leq L \int_0^t \|\mathbf{x}(s) - \mathbf{y}(s)\| \, \mathrm{d}s \implies \|\mathbf{x}(t) - \mathbf{y}(t)\| = 0 \quad \forall t \geq 0,$$

where the last implication follows from Gronwall's inequality. This concludes uniqueness. In general, for $\mathbf{x}(0) = \hat{\mathbf{x}}$ and $\mathbf{y}(0) = \hat{\mathbf{y}}$, we have

$$\begin{aligned} \|\mathbf{x}(t) - \mathbf{y}(t)\| &\leq \|\hat{\mathbf{x}} - \hat{\mathbf{y}}\| + L \int_0^t \|\mathbf{x}(s) - \mathbf{y}(s)\| \, \mathrm{d}s \\ \implies \|\mathbf{x}(t) - \mathbf{y}(t)\| &\leq e^{Lt} \|\hat{\mathbf{x}} - \hat{\mathbf{y}}\|, \end{aligned}$$

by the Gronwall's inequality implying continuous dependence on the initial condition. Hence, the ODE is well-posed. □

A few remarks:

1. Picard iteration is not a good computational scheme. In practice, Euler scheme is the most basic choice. Suppose h is bounded and let $a := \frac{T}{N}$ where $N \gg 1$. Now, let

   $$\mathbf{X}_N((n+1)a) := \mathbf{X}_N(na) + ah(\mathbf{X}_N(na)), \quad 0 \leq n < N.$$

   We interpolate linearly to get

   $$\mathbf{X}_N(t) := \mathbf{X}_N(na) + (t - na)h(\mathbf{X}_N(na)), \quad t \in [na, (n+1)a].$$

   Then, as $N \uparrow \infty$, $\mathbf{X}_N(t), t \in [0, T]$ converges to a solution of the ODE uniformly on $[0, T]$. This too proves the existence of a solution and needs only the continuity of h. However, uniqueness may fail. In numerical analysis, more sophisticated discretisations are available.

2. A local Lipschitz condition on h gives local well-posedness for a small time interval, but the solution may not exist for all time.

3. The linear growth condition shown below suffices for a solution to exist for all time:

   $$\|h(\mathbf{x}\| \leq K(1 + \|\mathbf{x}\|)$$

   for some $K > 0$. Then, we have

   $$\begin{aligned} \|\mathbf{x}(t)\| \leq \|\mathbf{x}(0)\| + \left\| \int_0^t h(\mathbf{x}(s) \, \mathrm{d}s \right\| &\leq \|\mathbf{x}(0)\| + \int_0^t K(1 + \|\mathbf{x}(s)\|) \, \mathrm{d}s \\ \implies \|\mathbf{x}(t)\| &\leq (\|\mathbf{x}(0)\| + KT)e^{Kt}, \quad t \in [0, T], \end{aligned}$$

   by Gronwall's inequality. We further note that the Lipschitz condition implies linear growth. A proof of this is left as an exercise for the reader.

4. A symmetric well-posedness theory can be developed for $t \leq 0$. Thus, for Lipschitz h, there is a unique solution for all $t \in \mathbb{R}$.

5. We also have a *discrete* Gronwall's inequality, which is proved similarly. Let $x_n \geq 0, a_n \geq 0, n \geq 0$, and $C, K > 0$ such that

$$x_{n+1} \leq C + K \sum_{m=0}^{n} a_m x_m \quad \forall n \geq 0.$$

Then, $x_{n+1} \leq C e^{K \sum_{m=0}^{n} a_m}$ for all $n \geq 0$.

We now take a more qualitative look at ODEs. Assume well-posedness. There are two broad ways of thinking about ODEs.

1. We can think of the ODE as the graph of $t \mapsto \mathbf{x}(t) \in \mathbb{R}^d$, that is, we think of $\mathbf{x}(t)$ as a function of time. The component-wise time derivative at t is $h(\mathbf{x}(t))$.

2. We can think of the ODE as a trajectory, or a curve $\mathbf{x}(\cdot)$ in $\mathbb{R}^d$ with t as a running parameter. This is also called a phase portrait. The tangent at point $\mathbf{x}$ on the curve is $h(\mathbf{x})$. One often flips this picture around and imagines a vector $h(\mathbf{x})$ at each point $\mathbf{x}$ (a vector field) and think of trajectories as curves drawn that are tangent to the vector field at all points (integral curves).

Definition 1.2.3: Limit Sets

The ω-limit set of a trajectory $\mathbf{x}(\cdot)$ is the set of all points $\mathbf{x}$ such that $\exists\, t_n$ *uparrow*∞ such that $\mathbf{x}(t_n) \to \mathbf{x}$, that is, the set of limit points of $\mathbf{x}(t)$ as $t \uparrow \infty$. One can show that this set is closed but can be empty. The α-limit set is defined similarly for $t_n \downarrow -\infty$.

Definition 1.2.4: Invariance

A set $A \subseteq \mathbb{R}^d$ is said to be *positively invariant* if $\mathbf{x}(0) \in A \implies \mathbf{x}(t) \in A \quad \forall t \geq 0$. Negative invariance is defined similarly. A set that is both positively and negatively invariant is said to be *invariant*.

Proposition 1.2.5

The ω- and α-limit sets are invariant.

Definition 1.2.6: Liapunov Stable

An equilibrium $\mathbf{x}^*$ is said to be *Liapunov stable* if given $\epsilon > 0$, there exists a $\delta > 0$ such that

$$\|\mathbf{x}(0) - \mathbf{x}^*\| < \delta \implies \|\mathbf{x}(t) - \mathbf{x}^*\| < \epsilon \quad \forall t \geq 0.$$

Definition 1.2.7: Asymptotically Stable

An equilibrium $\mathbf{x}^*$ is said to be *asymptotically stable* if it is Liapunov stable and there exists an open neighbourhood $\mathcal{O}$ of $\mathbf{x}^*$ such that

$$\mathbf{x}(0) \in \mathcal{O} \implies \mathbf{x}(t) \to \mathbf{x}^*.$$

Definition 1.2.8: Domain of Attraction

The largest positively invariant open set $\mathcal{D}$ such that

$$\mathbf{x}(0) \in \mathcal{D} \implies \mathbf{x}(t) \to \mathbf{x}^*$$

is called the *domain of attraction* of $\mathbf{x}^*$.

One sufficient condition for the above is that there exist a continuously differentiable $V \colon \mathcal{D} \to [0, \infty)$ such that

$$\lim_{\mathbf{x} \to \partial\mathcal{D}} V(\mathbf{x}) = \infty, \text{ and}$$

$$\langle \nabla V(\mathbf{x}), h(\mathbf{x}) \rangle < 0 \quad \forall \mathbf{x} \in \mathcal{D}, \mathbf{x} \neq \mathbf{x}^*.$$

Thus, we have

$$\frac{\mathrm{d}}{\mathrm{d}t} V(\mathbf{x}(t)) = \langle \nabla V(\mathbf{x}(t)), h(\mathbf{x}(t)) \rangle < 0 \quad \text{when } \mathbf{x}(t) \neq \mathbf{x}^*,$$

that is, V decreases along the trajectory. Since $V \geq 0$, $\mathbf{x}(t) \to \mathbf{x}^*$. Further, if we consider $\mathcal{B}_c(\mathbf{x}^*) := \{\mathbf{x} \mid V(\mathbf{x}) \leq c\} \subseteq \mathcal{D}$ for a suitable $c > V(\mathbf{x}^*$, then

$$\mathbf{x}(0) \in \mathcal{B}_c(\mathbf{x}^*) \implies \mathbf{x}(t) \in \mathcal{B}_c(\mathbf{x}^*) \quad \forall t \geq 0.$$

Note that $\mathbf{x}^* \in \mathcal{B}_c(\mathbf{x}^*)$ and $\mathcal{B}_c(\mathbf{x}^*)$ shrinks to $\{\mathbf{x}^*\}$ as $c \downarrow 0$. In particular, any ϵ-neighbourhood of $\mathbf{x}^*$ contains $\mathcal{B}_c(\mathbf{x}^*)$ for sufficiently small c. Thus, $\mathbf{x}^*$ is Liapunov stable and hence asymptotically stable. In this case, V is called a *Liapunov function*. Conversely, if $\mathbf{x}^*$ is asymptotically stable, then such a V exists and can be taken to satisfy $V(\mathbf{x}) \to \infty$ as $\mathbf{x} \to \partial\mathcal{D}$.

More generally, we have the *LaSalle Invariance Prinsciple* which states that $\mathbf{x}(t)$ converges to the largest invariant set contained in $\mathcal{A} := \{\mathbf{x} \mid \langle V(\mathbf{x}, h(\mathbf{x})\rangle = 0\}$.

If there exists some continuously differentiable $V \colon \mathcal{D} \to [0, \infty)$ such that

$$\lim_{\mathbf{x} \to \partial\mathcal{D}} V(\mathbf{x}) = \infty, \text{ and}$$
$$\langle \nabla V(\mathbf{x}), h(\mathbf{x})\rangle < 0 \quad \forall \mathbf{x} \notin \mathcal{C}$$

for some bounded set $\mathcal{C}$, then

$$\frac{\mathrm{d}}{\mathrm{d}t} V(\mathbf{x}(t)) = \langle \nabla V(\mathbf{x}(t)), h(\mathbf{x}(t))\rangle < 0 \quad \text{when } \mathbf{x}(t) \notin \mathcal{C},$$
$$\implies \mathbf{x}(t) \to \mathcal{C}.$$

In particular, the trajectories remain bounded.

We now consider the linear system $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t)$ for some $\mathbf{A} \in \mathbb{R}^{d \times d}$. Then the origin, $\mathbf{0}$, is an equilibrium. If $\mathbf{A}$ is non-singular, it is the only equilibrium. It is asymptotically stable if all eigenvalues of $\mathbf{A}$ are in the left half plane. If not, suppose there are no eigenvalues on the imaginary axis. Suppose there are $m < d$ eigenvalues in the left half plane. We can write $\mathbb{R}^d = \mathcal{S} \oplus \mathcal{U}$ where $\mathcal{S}$ is the m-dimensional stable subspace corresponding to the eigenvectors of eigenvalues in the left half plane, and $\mathcal{U}$ is the $(d-m)$-dimensional stable subspace corresponding to the eigenvectors of eigenvalues in the right half plane. Then, $\mathbf{x}(0) \in \mathcal{S} \implies \mathbf{x}(t) \to \mathbf{0}$ and $\mathbf{x}(0) \in \mathcal{U}$ implies that $\mathbf{x}(t)$ moves away from $\mathbf{0}$. More importantly, if $\mathbf{x}(0) \notin \mathcal{S}$, $\mathbf{x}(t)$ eventually moves away from $\mathbf{0}$. That is,

$$\mathbf{x}(t) \to \mathbf{0} \iff \mathbf{x}(0) \in \mathcal{S}.$$

However, $\mathcal{S}$ has zero volume in $\mathbb{R}^d$, and thus, for a typical initial condition, $\mathbf{x}(t)$ eventually moves away from $\mathbf{0}$. The above arguments extend to any point $\mathbf{x}^* \in \mathbb{R}^d$ if we replace the linear ODE by the following affine ODE:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\left(\mathbf{x}(t) - \mathbf{x}^*\right).$$

We now extend these ideas to the non-linear case. Suppose h is continuously differentiable and let $Dh(\mathbf{x})$ denote its Jacobian matrix at $\mathbf{x}$, that is, the (i,j) entry of of $Dh(\mathbf{x})$ is $\frac{\partial h_i}{\partial x_j}(\mathbf{x})$. By Taylor formula, for $\mathbf{x} \approx \mathbf{x}^*$, we have

$$h(\mathbf{x}) \approx h(\mathbf{x}^*) + Dh(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*) = Dh(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*).$$

We now consider the affine ODE

$$\dot{\mathbf{z}}(t) = Dh(\mathbf{x}^*)\left(\mathbf{z}(t) - \mathbf{x}^*\right)$$

which is called the *linearisation* of the original ODE at $\mathbf{x}^*$.

Theorem 1.2.9: Hartman-Großman Theorem

Let $\mathbf{x}^*$ be a *hyperbolic* equilibrium, that is, $Dh(\mathbf{x}^*)$ has no eigenvalues on the imaginary axis. Then, there exist open neighbourhoods $\mathcal{O}_1, \mathcal{O}_2$ of $\mathbf{x}^*$ such that the phase portrait of the original ODE in $\mathcal{O}_1$ and its linearisation in $\mathcal{O}_2$ can be mapped to each other by a continuous and continuously invertible transformation.

Thus, 'stable subspaces' morph into 'stable manifolds' and 'unstable subspaces' morph into 'unstable manifolds'.

§1.3. Convergence Analysis

Recall that our iteration is

$$\mathbf{x}(n+1) := \mathbf{x}(n) + a(n)\left[h(\mathbf{x}(n)) + M(n+1)\right], \quad n \geq 0.$$

Since we view $a(n)$ as a discrete time step, we define the *algorithmic time scale* as

$$t_0 := 0, \quad t_n := \sum_{m=0}^{n-1} a(m), \quad n \geq 0.$$

Now, we define $\overline{\mathbf{x}}(t)$, $t \in [0, \infty)$ as follows.

$$\begin{aligned} \overline{\mathbf{x}}(t_n) &:= \mathbf{x}(n) \quad \forall n \geq 0 \text{ and} \\ \overline{\mathbf{x}}(t) &:= \overline{\mathbf{x}}(t_n) + \left(\frac{t - t_n}{t_{n+1} - t_n}\right)(\overline{\mathbf{x}}(t_{n+1}) - \overline{\mathbf{x}}(t_n)) \quad \text{for } t \in [t_n, t_{n+1}]. \end{aligned}$$

That is, we linearly interpolate on $[t_n, t_{n+1}]$. Then, $\overline{\mathbf{x}}$ is continuous and piecewise linear. We assume stability, that is,

$$\sup_{n \geq 0} \|\mathbf{x}(n)\| < \infty \quad \text{a.s.}$$

Fix $T > 0$. We shall compare $\overline{\mathbf{x}}(\cdot)$ on a sliding window $[t, t+T]$ as $t \uparrow \infty$, with the ODE trajectory on the same interval that matches with it at the beginning of the interval, that is, with $\mathbf{x}^t(s)$, $s \in [t, t+T]$, satisfying

$$\dot{\mathbf{x}}^t(s) = h(\mathbf{x}(s)),\ s \in [t, t+T],\ \mathbf{x}^t(t) = \overline{\mathbf{x}}(t).$$

It suffices to consider $t = t_n$ for some $n \geq 0$ since for the general case there is a negligible additional error which can be easily handled.

Let $m(n) := \min\{k \geq n \colon t_k \geq t_n + T\}$. Then, $t_{m(n)} \approx t_n + T$. We shall compare $\overline{\mathbf{x}}(\cdot)$ and $\mathbf{x}^{t_n}(\cdot)$ on the interval $[t_n, t_{m(n)}]$. Now, define

$$\zeta(n) := \sum_{m=0}^{n-1} a(m)M(m+1), \quad n \geq 1.$$

This is a martingale, that is, $\mathbb{E}[\zeta(n+1) \mid \mathcal{F}_n] = \zeta(n)$ for all n. Also,

$$\begin{aligned}
\sum_n \mathbb{E}\left[\|\zeta(n+1) - \zeta(n)\|^2 \mid \mathcal{F}_n\right] &= \sum_n a(n)^2 \mathbb{E}\left[\|M(n+1)\|^2 \mid \mathcal{F}_n\right] \\
&\leq \sum_n a(n)^2 K(1 + \|\mathbf{x}(n)\|^2) \\
&\leq K(1 + \sup_n \|\mathbf{x}(n)\|^2) \cdot \sum_m a(m)^2 \\
&< \infty \quad \text{a.s.}
\end{aligned}$$

By the convergence theorem for square integrable martingales, $\zeta(n)$ converges almost surely as $n \uparrow \infty$. We now define this convergence theorem more formally.

Theorem 1.3.1

Let $(Z_n, \mathcal{F}_n)$ be a square-integrable martingale. Its *quadratic variation process* $\langle Z \rangle_n$, $n \geq 0$ is given by

$$\langle Z \rangle_n := \sum_{m=0}^{n} \mathbb{E}\left[(Z_{m+1} - Z_m)^2 \mid \mathcal{F}_n\right].$$

Then, almost surely,

$$\lim_{n \uparrow \infty} \langle Z \rangle_n < \infty \implies \langle Z \rangle_n \text{ converges.}$$

Now, let $0 \leq k \leq m(n) - n$. Then,

$$\begin{aligned}
\overline{\mathbf{x}}(t_{n+k}) &= \overline{\mathbf{x}}(t_n) + \sum_{i=0}^{k-1} a(n+i) h(\overline{\mathbf{x}}(t_{n+i})) + \sum_{l=0}^{k-1} a(n+l) M(n+l+1) \\
&= \overline{\mathbf{x}}(t_n) + \sum_{i=0}^{k-1} a(n+i) h(\overline{\mathbf{x}}(t_{n+i})) + \zeta(n+k) - \zeta(n).
\end{aligned}$$

Further, we have

$$\begin{aligned}\mathbf{x}^{t_n}(t_{n+k}) &= \mathbf{x}^{t_n}(t_n) + \int_{t_n}^{t_{n+k}} h(\mathbf{x}^{t_n}(s))\,\mathrm{d}s \\ &= \mathbf{x}^{t_n}(t_n) + \sum_{i=0}^{k-1} a(n+i)\cdot h(\mathbf{x}^{t_n}(t_{n+i})) + \sum_{l=n}^{n+k-1}\int_{t_l}^{t_{l+1}} \left(h(\mathbf{x}^{t_n}(s)) - h(\mathbf{x}^{t_n}(t_l))\right)\,\mathrm{d}s\end{aligned}$$

because for $l \geq n$, we have

$$a(l)h(\mathbf{x}^{t_n}(t_l)) = \int_{t_l}^{t_{l+1}} h(\mathbf{x}^{t_n}(t_l))\,\mathrm{d}s.$$

Thus,

$$\mathbf{x}^{t_n}(t_{n+k}) = \mathbf{x}^{t_n}(t_n) + \sum_{i=0}^{k-1} a(n+i)\cdot h(\mathbf{x}^{t_n}(t_{n+i})) + \int_{t_n}^{t_{n+k}} \left(h(\mathbf{x}^{t_n}(s)) - h(\mathbf{x}^{t_n}([t]))\right)\,\mathrm{d}s$$

where $[t] := \max\{t_m \colon t_m \leq t\}$. Note that $\overline{\mathbf{x}}(t_n) = \mathbf{x}^{t_n}(t_n) = \mathbf{x}(n)$. Thus, we have

$$\left\|\overline{\mathbf{x}}(t_{n+k}) - \mathbf{x}^{t_n}(t_{n+k})\right\| \leq \sum_{i=0}^{k-1} a(n+i)\left\|h(\overline{\mathbf{x}}(t_{n+i})) - h(\mathbf{x}^{t_n}(t_{n+i}))\right\| + \mathcal{I}_d + \mathcal{I}_n$$

where $\mathcal{I}_d$ denotes the error due to discretisation and $\mathcal{I}_n$ denotes the error due to noise.

$$\left\|\overline{\mathbf{x}}(t_{n+k}) - \mathbf{x}^{t_n}(t_{n+k})\right\| \leq L\sum_{i=0}^{k-1} a(n+i)\left\|\overline{\mathbf{x}}(t_{n+i}) - \mathbf{x}^{t_n}(t_{n+i})\right\| + \mathcal{I}_d + \mathcal{I}_n$$

By the discrete Gronwall inequality, there exists $C(T) > 0$ such that

$$\sup_{n \leq m \leq m(n)} \left\|\overline{\mathbf{x}}(t_m) - \mathbf{x}^{t_n}(t_m)\right\| \leq C(T)(\mathcal{I}_d + \mathcal{I}_n)$$

For $\infty > K \geq \sup_{t\in[t_n,t_{m(n)}]} \|h(\mathbf{x}^{t_n}(t))\| > 0$, we have

$$\begin{aligned}\left\|\int_{t_m}^{t_{m+1}} \left(h(\mathbf{x}^{t_n}(s)) - h(\mathbf{x}^{t_n}([t]))\right)\,\mathrm{d}s\right\| &= \left\|\int_{t_m}^{t_{m+1}} \left(h(\mathbf{x}^{t_n}(s)) - h(\mathbf{x}^{t_n}(t_m))\right)\,\mathrm{d}s\right\| \\ &\leq L\int_{t_m}^{t_{m+1}} \left\|\mathbf{x}^{t_n}(s) - \mathbf{x}^{t_n}(t_m)\right\|\,\mathrm{d}s \\ &\leq L\int_{t_m}^{t_{m+1}} \left\|\int_{t_m}^{s} h(\mathbf{x}^{t_n}(u))\,\mathrm{d}u\right\|\,\mathrm{d}s \\ &\leq \frac{LK}{2}(t_{m+1} - t_m)^2 \\ &= L'a(m)^2.\end{aligned}$$

Hence,

$$\mathcal{I}_d = \left\| \int_{t_m}^{t_{m+1}} \left(h(\mathbf{x}^{t_n}(s)) - \overline{h}(\mathbf{x}^{t_n}([t])) \right) \, \mathrm{d}s \right\| \leq L' \sum_{m \geq n} a(m)^2 \downarrow 0 \text{ as } n \uparrow \infty.$$

Also,

$$\mathcal{I}_n \leq \sup_{m \geq 0} \|\zeta(n+m) - \zeta(n)\| \to 0 \text{ a.s. as } n \uparrow \infty.$$

Thus, as $n \uparrow \infty$,

$$\begin{aligned} &\max_{n \leq m \leq m(n)} \left\| \overline{\mathbf{x}}(t_m) - \mathbf{x}^{t_n}(t_m) \right\| \to 0 \text{ a.s. } \Longrightarrow \\ &\lim_{t \uparrow \infty} \max_{s \in [0,T]} \left\| \overline{\mathbf{x}}(t+s) - \mathbf{x}^{t_n}(t+s) \right\| \to 0 \text{ a.s. } \quad \forall T > 0. \end{aligned}$$

Now, let

$$\begin{aligned} \mathcal{D} :&= \left\{ \mathbf{x} \in \mathbb{R}^d \mid \exists\, 0 < s_n \uparrow \infty \text{ such that } \overline{\mathbf{x}}(s_n) \to \mathbf{x} \right\} \\ &= \left\{ \mathbf{x} \in \mathbb{R}^d \mid \exists\, 0 < t_k \uparrow \infty \text{ such that } \overline{\mathbf{x}}(t_k) \to \mathbf{x} \right\} \end{aligned}$$

Proposition 1.3.2

$\mathcal{D}$ is an invariant set for the ODE.

Proof. Suppose $s_n \uparrow \infty$ and $\overline{\mathbf{x}}(s_n) \to \mathbf{x}$. Then, $\mathbf{x} \in \mathcal{D}$. Let $\dot{\tilde{\mathbf{x}}}(t) = h(\tilde{\mathbf{x}}(t))$, $\tilde{\mathbf{x}}(0) = \mathbf{x}$. By the above, for $T > 0$, we have

$$\overline{\mathbf{x}}(s_n + T) - \mathbf{x}^{s_n}(s_n + T) \to 0.$$

By continuous dependence on initial condition,

$$\mathbf{x}^{s_n}(s_n) = \overline{\mathbf{x}}(s_n) \to \mathbf{x} \implies \mathbf{x}^{s_n}(s_n + T) - \tilde{\mathbf{x}}(T) \to 0.$$

Thus, $\mathbf{x}^{s_n}(s_n + T) - \tilde{\mathbf{x}}(T) \to 0$, implying $\tilde{\mathbf{x}}(T) \in \mathcal{D}$. Similar argument works for $T < 0$. Hence, $\mathcal{D}$ is invariant. □

Definition 1.3.3: Internally Chain Transitive

We say that $\mathcal{D}$ is an *internally chain transitive* invariant set if given any $\epsilon, T > 0$ and points $\mathbf{x}, \mathbf{y} \in \mathcal{D}$, we can find $n \geq 1$, and a chain of points $\mathbf{x} = \mathbf{x}_0, \ldots, \mathbf{x}_n = \mathbf{y}$ such that for $0 \leq i < n$, there exists a trajectory segment of the ODE of duration at least T which starts in the ϵ-neighbourhood of $\mathbf{x}_i$ and ends in the ϵ-neighbourhood of $\mathbf{x}_{i+1}$.

Theorem 1.3.4: Benaim's Theorem

$\mathbf{x}(n)$ converges to an internally chain transitive invariant set of the ODE (a.s.).

§1.4. Variants of the Robbins-Monro Algorithm

We mention several useful variations of the Robbins-Monro algorithm as well as several of its applications. For this section, we forego long proofs and only highlight some important results.

§§1.4.1. Two Time Scale Algorithm I

Here, we consider the coupled iterations

$$\begin{aligned}\mathbf{x}_{n+1} &= \mathbf{x}_n + a(n)\left[h(\mathbf{x}_n, \mathbf{y}_n) + M(n+1)\right] \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + b(n)\left[g(\mathbf{x}_n, \mathbf{y}_n) + M'(n+1)\right]\end{aligned}$$

with the modified Robbins-Monro conditions

$$\sum_n a(n) = \sum_n b(n) = \infty, \quad \sum_n (a(n)^2 + b(n)^2) < \infty, \qquad \frac{b(n)}{a(n)} \to 0.$$

Note that $\frac{b(n)}{a(n)} \to 0$ implies that $\{\mathbf{y}_n\}$ is updated on a slower timescale than $\{\mathbf{x}_n\}$. We may however, consider an 'algorithmic time scale' in terms of $\{a(n)\}$ and then rewrite the second iteration as

$$\mathbf{y}_{n+1} = \mathbf{y}_n + a(n)\left(\frac{b(n)}{a(n)}\right) \cdot \left[g(\mathbf{x}_n, \mathbf{y}_n) + M'(n+1)\right].$$

The limiting ODE for this is

$$\dot{\mathbf{x}}(t) = h(\mathbf{x}(t), \mathbf{y}(t)), \quad \dot{\mathbf{y}}(t) = 0,$$

that is, $\{\mathbf{x}_n\}$ sees $\{\mathbf{y}_n\}$ as quasi-static, or nearly constant. Thus, it tracks the ODE

$$\dot{\mathbf{x}}(t) = h(\mathbf{x}(t), \mathbf{y}),$$

for $\mathbf{y} \approx \mathbf{y}_n$. Suppose $\mathbf{x}(t) \to \lambda(\mathbf{y})$. Then, $\mathbf{x}_n - \lambda(\mathbf{y}_n) \to \mathbf{0}$ almost surely, that is, $\{\mathbf{y}_n\}$ sees $\{\mathbf{x}_n\}$ as quasi-equilibrated.

We now rewrite the for $\{\mathbf{y}_n\}$ as

$$\mathbf{y}_{n+1} = \mathbf{y}_n + b(n)\left[g(\lambda(\mathbf{y}_n), \mathbf{y}_n) + (g(\mathbf{x}_n, \mathbf{y}_n) - g(\lambda(\mathbf{y}_n), \mathbf{y}_n)) + M'(n+1)\right]$$

Using the algorithmic time scale defined in terms of $\{b(n)\}$, we see that $\{\mathbf{y}_n\}$ tracks the ODE

$$\dot{\mathbf{y}}(t) = g\left(\lambda(\mathbf{y}(t)), \mathbf{y}(t)\right).$$

If $\mathbf{y}(t) \to \mathbf{y}^*$, then $(\mathbf{x}_n, \mathbf{y}_n) \to (\lambda(\mathbf{y}^*), \mathbf{y}^*)$. This is very similar to *singularly perturbed differential equations*:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= h(\mathbf{x}(t), \mathbf{y}(t)) \\ \dot{\mathbf{y}}(t) &= \epsilon \cdot g(\mathbf{x}(t), \mathbf{y}(t))\end{aligned}$$

in the $\epsilon \downarrow 0$ limit. This scheme also emulates nested iterations. This is extremely for many reinforcement learning algorithms such as actor-critic methods, as well as policy gradient method.

§§1.4.2. Two Time Scale Algorithm II

Here, we consider the Robbins-Monro algorithm in the presence of Markov noise $\{Y_n\}$. We consider the iteration

$$\mathbf{x}_{n+1} = \mathbf{x}_n + a(n)\left[h(\mathbf{x}_n, Y_n) + M(n+1)\right],$$

where

$$\mathbb{P}\left(Y_{n+1} \in A \mid \mathcal{F}_n\right) = p_{\mathbf{x}_n}(A \mid Y_n).$$

Let $p_{\mathbf{x}}(\cdot \mid \cdot)$ be irreducible with unique stationary distribution $\pi_{\mathbf{x}}$. Then, $\{\mathbf{x}_n\}$ tracks the ODE

$$\dot{\mathbf{x}}(t) = \int h(\mathbf{x}(t), y) \cdot \pi_{\mathbf{x}(t)}(\mathrm{d}y).$$

The above notation just means that we are taking an expectation over y. Intuitively, $\{\mathbf{x}_n\}$ is updated on a slow time scale. Similar to the previous algorithm, the slower time scale iterate tracks the asymptotic behaviour of the faster time scale iterate. Here, however, the faster process is a Markov chain, and does not converge. What we have instead, is that it converges to a stationary distribution, and the resulting ODE tracks just the expected value of $h(\mathbf{x}(t), y)$ with respect to the stationary distribution of y. As before, $\{\mathbf{x}_n\}$ sees $\{Y_n\}$ as quasi-equilibrated. Here, however, quasi-equilibrated does not mean that $\{Y_n\}$ converges to a definite value. Rather, $\{Y_n\}$ equilibrates at some particular stationary distribution.

§§1.4.3. Two Time Scale Algorithm III

This algorithm was proposed by Tstitsiklis-Bertsekas-Athans, and deals with distributed optimisation. This is also called a gossip algorithm. Here, processor i

performs the vector iteration

$$\mathbf{x}_{n+1}^i = \sum_j p(j \mid i) \cdot \mathbf{x}_n^j + a(n) \left[h^i(\mathbf{x}_n^i) + M^i(n+1) \right]$$

where $P := [[p(\cdot \mid \cdot)]]$ is an irreducible stochastic matrix with stationary distribution π. An important case is when P is doubly stochastic matrix, where we have π being uniform. For any arbitrary stochastic matrix, we may convert it to a doubly stochastic matrix by applying the *Sinkhorn-Knopp algorithm* that alternatively normalises rows and columns until convergence. Intuitively, each processor takes a weighted average of information from all other processors to update its current iterate. This scheme tracks the ODE

$$\dot{\mathbf{x}}^i(t) = \sum_j \pi(j) \cdot h^j(\mathbf{x}(t)).$$

Also, we have $\|\mathbf{x}_n^i - \mathbf{x}_n^j\| \to 0$ almost surely. Thus, we have convergence to a common limit. That is, all the processors do exactly the same thing asymptotically.

§§1.4.4. Stability Test

Suppose that $h_\infty(\mathbf{x}) := \lim_{c \uparrow \infty} \dfrac{h(c\mathbf{x})}{c}$ exists and the ODE

$$\dot{\mathbf{x}}(t) = h(\mathbf{x}(t))$$

has the origin as its unique globally asymptotically stable equilibrium. Then, $\sup_n \|\mathbf{x}_n\| < \infty$ almost surely. There are other tests possible as well. The intuition is that if $\|\mathbf{x}_{n_k}\| \uparrow \infty$, then the rescaled interpolations on $[t_{n_k}, t_{m(n_k)}]$ given by

$$\check{\mathbf{x}}(t) := \frac{\overline{\mathbf{x}}(t)}{\|\overline{\mathbf{x}}(t_{n_k})\|}, \quad t \in [t_{n_k}, t_{m(n_k)}],$$

track the ODE $\dot{\mathbf{x}}(t) = h(\mathbf{x}(t))$ and hence tend to the origin. Since $\overline{\mathbf{x}}(\cdot)$ and $\check{\mathbf{x}}(\cdot)$ differ only by a scale factor, the same holds true for $\overline{\mathbf{x}}(\cdot)$. Hence, the iterates cannot blow up.

§§1.4.5. Constant Stepsize Algorithm

Here, we consider the constant stepsize $a(n) \equiv a$ for all $n \geq 0$. Here we have weaker claims. We have 'high probability concentration' rather than 'almost sure convergence'. We have

$$\limsup_{n \uparrow \infty} \mathbb{E}\left[\|\mathbf{x}_n - \mathbf{x}^*\|^2 \right] \leq Ka$$

This algorithm is particular useful for slowly varying environments. For decreasing stepsize, the algorithmic time scale eventually becomes slower than the environment and therefore cannot track it. This is also useful when the algorithm is hardwired.

§§1.4.6. Distributed and Asynchronous Iterates

Here, we consider the iteration

$$\mathbf{x}_{n+1}(i) = \mathbf{x}_n(i) + a(n) \cdot \mathbb{I}\{i \in S_n\} \left[h_i \left(\mathbf{x}_{n-\tau_{1i}(n)}(1), \ldots, \mathbf{x}_{n-\tau_{di}(n)}(d)\right) + M_{n+1}(i)\right]$$

The argument of the iterate denotes its component as well as its processor. Here $S_n \subseteq \{1, \ldots, d\}$ is the set of indices of the components updated at time n. $\tau_{ji}(n) := n-$ the time stamp of the most recent value received by i from j. For bounded delays (or with a conditional moment bound), the delays do not matter asymptotically as the time scale $(n \to t(n) := \sum_{m=0}^{n} a(m))$ is getting shrunk.

This scheme tracks the ODE

$$\dot{\mathbf{x}}(t) = \Lambda(t) h(\mathbf{x}(t)),$$

where $\Lambda(t)$ is a diagonal matrix with non-negative diagonal entries $\lambda_i(t)$ reflecting relative frequencies of updates of different components. For example, in the TD(λ) algorithm, we update just one component at a time. This is neatly represented with $S_n = \{X_n\}$ where $\{X_n\}$ is an ergodic Markov chain on $\{1, \ldots, d\}$ with stationary distribution π. The i^{th} diagonal entry is simply $\pi(i)$.

Now, we replace $a(n)$ by $a(\nu(i, n))$ where

$$\nu(i, n) := \sum_{m=0}^{n} \mathbb{I}\{i \in S_m\}$$

denotes the 'local clock' that counts the number of times component i is updated till time n. Then, under some further conditions on $\{a(n)\}$, we have that $\Lambda(t) = \alpha(t)\mathbf{I}$ asymptotically. For example, the algorithmic time scale for components i and j are

$$\sum_{m=0}^{n} a(\nu(i, m)) \quad \text{and} \quad \sum_{m=0}^{n} a(\nu(j, m)) \quad \text{respectively.}$$

For $a(n) = \frac{1}{n+1}$, we have $\sum_{m=0}^{n} \frac{1}{m+1} \approx \log n$. Hence, if

$$\liminf_{n \uparrow \infty} \frac{\nu(i, n)}{n} > 0 \quad \text{a.s.},$$

then we have

$$\begin{aligned}\lim_{n\uparrow\infty}\frac{\sum_{m=0}^{n}a(\nu(i,m))}{\sum_{m=0}^{n}a(\nu(j,m))} &= \lim_{n\uparrow\infty}\frac{\log\nu(i,n)}{\log\nu(j,n)}\\ &= \lim_{n\uparrow\infty}\frac{\log\frac{\nu(i,n)}{n}+\log n}{\log\frac{\nu(j,n)}{n}+\log n}\\ &= 1.\end{aligned}$$

For some special algorithms (such as stochastic gradient descent, $\|\cdot\|_\infty$-contractions), we have that

$$\liminf_{n\uparrow\infty}\frac{\nu(i,n)}{n}>0 \quad \text{a.s.} \,(\implies \liminf\lambda_i(t)>0 \quad \text{a.s.})$$

ensures correct convergence. For stochastic gradient descent in particular, we have $h(\mathbf{x})=-\nabla f(\mathbf{x})$ and

$$\frac{\mathrm{d}f(\mathbf{x}(t))}{\mathrm{d}t}=-\sum_i\lambda_i(t)\left(\frac{\partial f}{\partial x_i}(\mathbf{x}(t))\right)^2<0$$

except when $\nabla f(\mathbf{x})=\mathbf{0}$.

For finding fixed points of $\|\cdot\|_\infty$-contractions F, we set $h(\mathbf{x})=F(\mathbf{x})-\mathbf{x}$, where

$$\|F(\mathbf{x})-F(\mathbf{y})\|_\infty\le\alpha\cdot\|\mathbf{x}-\mathbf{y}\|_\infty \quad \text{for some } \alpha\in(0,1).$$

By the contraction mapping theorem, there exists a unique fixed point $\mathbf{x}^*$ satisfying $F(\mathbf{x}^*)=\mathbf{x}^*$. If $\lambda_i(t)\ge\delta>0$ for all t,i, then

$$\tilde{F}_t(\mathbf{x}):=(\mathbf{I}-\Lambda(t))\mathbf{x}+\Lambda(t)F(\mathbf{x})$$

satisfies: $\tilde{F}_t$ is an $\|\cdot\|_\infty$-contraction with contraction factor $(1-\delta(1-\alpha))$ and $\mathbf{x}^*$ is the unique fixed point of $\tilde{F}_t$ for all t. The limiting ODE is then

$$\dot{\mathbf{x}}(t)=\Lambda(t)(F(\mathbf{x}(t))-\mathbf{x}(t))=\tilde{F}_t(\mathbf{x}(t))-\mathbf{x}(t).$$

Thus, $\mathbf{x}(t)\to\mathbf{x}^*$ as desired. This scheme is particularly useful in reinforcement learning.

§§1.4.7. Stochastic Recursive Inclusions

Here we consider the iteration

$$\mathbf{x}_{n+1}=\mathbf{x}_n+a(n)\left[\mathbf{y}_n+M_{n+1}\right]$$

where $\mathbf{y}_n \in F(\mathbf{x}_n)$ for some **Marchaud** map

$$\mathbf{x} \in \mathbb{R}^d \mapsto F(\mathbf{x}) \subset \mathbb{R}^d,$$

that is, which satisfies

1. $\forall\, \mathbf{x}$, $F(\mathbf{x})$ is closed, bounded, and convex.
2. F has a closed graph, that is, the set $\{(\mathbf{x}, \mathbf{y}) \mid \mathbf{y} \in F(\mathbf{x})\}$ is closed. Equivalently,
$$(\mathbf{x}_n, \mathbf{y}_n) \to (\mathbf{x}, \mathbf{y}),\ \mathbf{y}_n \in F(\mathbf{x}_n) \forall\, n \implies \mathbf{y} \in F(\mathbf{x}).$$
3. F has at most linear growth: $\exists K > 0$ such that
$$\mathbf{y} \in F(\mathbf{x}) \implies \|\mathbf{y}\| \leq K(1 + \|\mathbf{x}\|).$$

The iteration tracks the differential inclusion

$$\dot{\mathbf{x}}(t) \in F(\mathbf{x}(t)).$$

For example, we have the stochastic subgradient descent, where $F = \partial f$, the subgradient of a convex function f at $\mathbf{x}$, defined as the cone

$$\partial f(\mathbf{x}) := \left\{\mathbf{y} \in \mathbb{R}^d \mid f(\mathbf{z}) \geq f(\mathbf{x}) + \langle \mathbf{y}, \mathbf{z} - \mathbf{x}\rangle\ \forall \mathbf{z}\right\}.$$

§1.5. Applications

§§1.5.1. Stochastic Gradient Descent

Here we consider $h(\mathbf{x}) = -\nabla f(\mathbf{x})$, and we track the ODE

$$\dot{\mathbf{x}}(t) = -\nabla f(\mathbf{x}(t)).$$

With 'rich noise', we have almost sure convergence to a local minimum if equilibria are isolated. (In general, pointwise convergence is not obvious but it holds for real analytic f). But, any isolated local minimum is a possible equilibrium with positive probability. This iterative scheme can slow down near saddle points. In this case, we can use momentum to accelerate the algorithm. That is, we use the iteration

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \underbrace{a(n)\left[-\nabla f(\mathbf{x}_n) + M_{n+1}\right]}_{\text{stochastic gradient term}} + \underbrace{b(n)\left[\mathbf{x}_n - \mathbf{x}_{n-1}\right]}_{\text{momentum term}}.$$

We may rewrite this as

$$\begin{aligned}\mathbf{x}_{n+1} &= \mathbf{x}_n + \mathbf{y}_n \\ \mathbf{y}_{n+1} &= \mathbf{y}_n - (1 - b(n))\mathbf{y}_n + a(n)\left[-\nabla f(\mathbf{x}_n) + M_{n+1}\right].\end{aligned}$$

This is reminiscent of the second order ODE

$$\dot{\mathbf{x}}(t) = \mathbf{y}(t), \quad \dot{\mathbf{y}}(t) = -\alpha \mathbf{y}(t) - \nabla f(\mathbf{x}(t)).$$

Intuitively, this momentum scheme emulates Newton's laws with friction in a potential well. This allows the iterate to speed up near unstable critical points and on flat landscapes, and allows it to escape shallow valleys. This scheme helps in neural network training since the composition of sigmoidal functions often leads to flat patches. Note that we are not claiming that the iterate converges to the global minimum. It is entirely possible that there is a deep enough local minimum such that the momentum of the iterate is not enough for it to escape this minimum. However, the iterate automatically avoids irrelevant shallow minima. Moreover, this method avoids zigzagging in 'pinched' landscapes, such as the rain-gutter landscape, shown below in Figure 1.1.

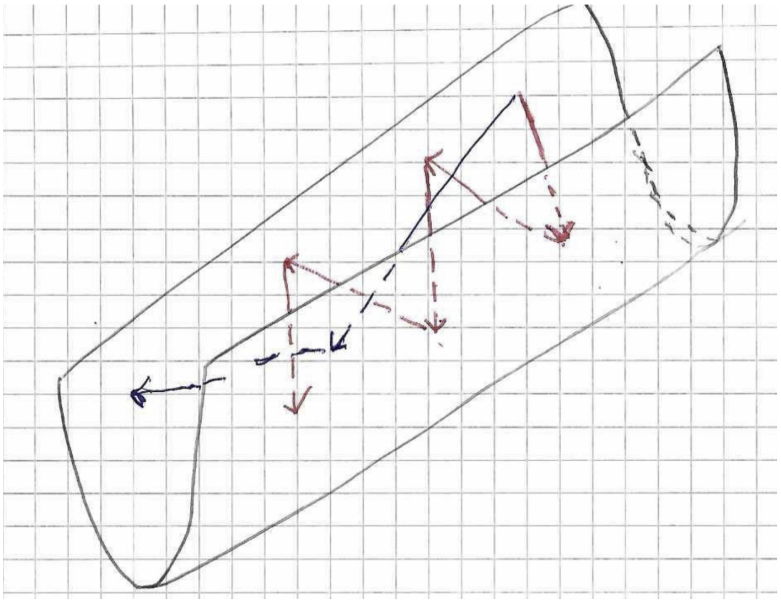

Figure 1.1: An example of a 'pinched' landscape often called a rain-gutter landscape. Using traditional stochastic gradient descent, our iterate zigzags along the landscape (maroon trajectory), whereas adding momentum avoids this phenomenon (black trajectory)

There are more sophisticated variants such as Nesterov's accelerated gradient method.

Typically, we do not have access to the gradient explicitly, and we must use an approximation. Sometimes, we may also be reluctant to evaluate the function too often since the evaluation procedure may be expensive. We now look at some approximation schemes.

1. **Kiefer-Wolfowitz.** This scheme uses finite difference approximations as:

$$\frac{\partial f}{\partial x_i}(\mathbf{x}) \approx \frac{f(\mathbf{x}+\delta\mathbf{e}_i)-f(\mathbf{x})}{\delta}$$
$$\frac{\partial f}{\partial x_i}(\mathbf{x}) \approx \frac{f(\mathbf{x}+\delta\mathbf{e}_i)-f(\mathbf{x}-\delta\mathbf{e}_i)}{2\delta}$$

   These require $d+1$ and $2d$ function evaluations respectively. The discretisation error in the latter is better but it requires more function evaluations.

2. **Simultaneous perturbation (Spall).** We take $\Delta_n(i)$ ($n \geq 0$, $1 \leq i \leq d$) to be i.i.d. ± 1 with equal probability. We set $\Delta_n := (\Delta_n(1), \ldots, \Delta_n(d))$. Then,

$$\begin{aligned}\frac{\partial f}{\partial x_i}(\mathbf{x}) &\approx \frac{f(\mathbf{x}+\delta\Delta_n)-f(\mathbf{x})}{\delta\Delta_n(i)} \\ &\approx \frac{\partial f}{\partial x_i}(\mathbf{x}) + \sum_{j\neq i} \frac{\partial f}{\partial x_j}\frac{\Delta_n(j)}{\Delta_n(i)}.\end{aligned}$$

   The second term can be absorbed into M_{n+1}. We may also use a single function estimate as follows.

$$\begin{aligned}\frac{\partial f}{\partial x_i}(\mathbf{x}) &\approx \frac{f(\mathbf{x}+\delta\Delta_n)}{\delta\Delta_n(i)} \\ &= \frac{f(\mathbf{x})}{\delta\Delta_n(i)} + \frac{\partial f}{\partial x_i}(\mathbf{x}) + \sum_{j\neq i} \frac{\partial f}{\partial x_j}\frac{\Delta_n(j)}{\Delta_n(i)}.\end{aligned}$$

   Here, both the first and third term can be absorbed into M_{n+1}. This scheme has numerical issues for small δ (small divisor problem).

3. **Flaxman-Kalai-McMahan.** We pick $\xi_n \in \mathbb{R}^d$ to be i.i.d. with zero mean

and identity covariance matrix. Then,

$$\begin{aligned} f_i(\mathbf{x}+\delta\xi_n)\xi_n(i) &\approx f_i(\mathbf{x})\xi_n(i) + \delta\frac{\partial f}{\partial x_i}(\mathbf{x})\xi_n(i)^2 + \delta\sum_{j\neq i}\frac{\partial f}{\partial x_i}(\mathbf{x})\xi_n(j)\xi_n(i) \\ &= \delta\frac{\partial f}{\partial x_i}(\mathbf{x}) + f_i(\mathbf{x})\xi_n(i) + \delta\frac{\partial f}{\partial x_i}(\mathbf{x})(\xi_n(i)^2-1) + \delta\sum_{j\neq i}\frac{\partial f}{\partial x_i}(\mathbf{x})\xi_n(j)\xi_n(i) \\ &= \delta\frac{\partial f}{\partial x_i}(\mathbf{x}) + \text{martingale difference terms.} \end{aligned}$$

4. **Mukherjee-Zhao.** To evaluate the gradient at $\mathbf{x}_0$, we recall that for $\mathbf{x}$ in a neighbourhood of $\mathbf{x}_0$, we have

$$f(\mathbf{x}) \approx f(\mathbf{x}_0) + \langle \nabla f(\mathbf{x}_0), \mathbf{x}-\mathbf{x}_0\rangle\,.$$

   We sample points $\mathbf{x}_i$, $1 \le i \le n$, in a neighbourhood of $\mathbf{x}_0$. Then,

$$\nabla f(\mathbf{x}_0) \approx \arg\min_{\mathbf{y}} \sum_{m=1}^{n} \mathcal{K}(\mathbf{x}_m, \mathbf{x}_0)\cdot\left(f(\mathbf{x}_m) - f(\mathbf{x}_0) - \langle \mathbf{y}, \mathbf{x}-\mathbf{x}_0\rangle\right)^2,$$

   where $\mathcal{K}\colon \mathbb{R}^d \times \mathbb{R}^d \to [0,\infty)$ is a suitable kernel function that satisfies $\mathcal{K}(\mathbf{x},\mathbf{y}) \to 0$ as $\|\mathbf{x}-\mathbf{y}\| \to \infty$.

5. **Katkovnik-Kulchitsky.** Consider a Gaussian $g_\sigma(\cdot)$ with zero mean and variance σ^2. Then, for small σ, we can express $f(x)$ as a convolution with g_σ. That is,

$$f(x) \approx \int g_\sigma(x-y)\cdot f(y)\,\mathrm{d}y.$$

   This gives us

$$\nabla f(x) \approx \int \nabla g_\sigma(x-y)\cdot f(y)\,\mathrm{d}y,$$

   which is still a Gaussian expectation and can be estimated using Monte Carlo simulation. This scheme is particularly useful when f is not differentiable to begin with. This scheme again has the small divisor problem for small σ, which can be ameliorated by additional averaging.

We also have a simulation-based optimisation paradigm, where we want to minimise $\mathbb{E}_\theta[f(X)]$ (X is a real valued random variable) over θ where θ parameterises the distribution of X. We generate i.i.d. $X_n = \Phi(U_n,\theta)$ with the law corresponding to θ with $\{U_n\}$ uniform on $[0,1]$. Suppose Φ is continuously differentiable. We do

$$\theta_{n+1} = \theta_n - a(n)\cdot\frac{\partial}{\partial\theta} f\left(\Phi(U_{n+1},\theta)\right)\Big|_{\theta=\theta_n}.$$

A more sophisticated version is the likelihood ratio method, due to Glynn. Suppose the distributions P_θ corresponding to θ have densities with respect to a base distribution P_{θ_0}. Let Λ_θ denote the likelihood ratio of P_θ with respect to P_{θ_0}. Then, $\mathbb{E}_\theta[f(X)] = \mathbb{E}_{\theta_0}[f(X)\Lambda_\theta(X)]$. The algorithm then is

$$\theta_{n+1} = \theta_n - a(n) \cdot f(X_{n+1}) \cdot \nabla_\theta \Lambda_\theta(X_{n+1})\Big|_{\theta=\theta_n}.$$

For global minimisation, we use the simulated annealing method, given by

$$\mathbf{x}_{n+1} = \mathbf{x}_n + a(n)\left[-\nabla f(\mathbf{x}_n) + M_{n+1}\right] + b(n)W_{n+1}$$

where $b(n) > 0$ is chosen appropriately given $\{a(n)\}$, and $\{W_n\}$ are i.i.d. $\mathcal{N}(0,1)$. This tracks the stationary distribution of the stochastic differential equation

$$\mathrm{d}X(t) = -\nabla f(X(t))\,\mathrm{d}t + \frac{C}{\log T}\,\mathrm{d}B(t)$$

as $T = t \uparrow \infty$, which asymptotically concentrates on the global minima of f. If we choose $a(n) = \frac{1}{n}$, then we choose $b(n) = \frac{C}{\sqrt{n \log\log n}}$.

§§1.5.2. Stochastic Gradient Descent for Machine Learning

Here, we are given samples of input-output pairs $(\mathbf{X}_n, \mathbf{Y}_n)$, $n \geq 1$. We wish to minimise the 'empirical risk', given by

$$\frac{1}{N}\sum_{m=1}^{N} \mathcal{L}(\mathbf{X}_m, \mathbf{Y}_m, \Theta) \approx \mathbb{E}_\Theta[\mathcal{L}(\mathbf{X}_n, \mathbf{Y}_n, \Theta)].$$

To replace the expectation by an average, we need the uniform strong law of large numbers. Sufficient conditions for these are given by the Vapnik-Chervonenkis theory and its extensions. Our target is to get sufficiently close to the minimum risk. That is, we want

$$\mathbb{E}\left[\mathcal{L}(\mathbf{X}_n, \mathbf{Y}_n, \Theta_n)\right] \leq \min_\Theta \mathbb{E}\left[\mathcal{L}(\mathbf{X}_n, \mathbf{Y}_n, \Theta)\right] + \epsilon$$

for $0 < \epsilon \ll 1$.

There are many special purpose variants of this, such as mini-batch algorithms, ADAGRAD, ADAM. Sometimes, the number of samples is large enough to be comparable to the ambient space. A typical fix for this is to use block-coordinate descent.

§§1.5.3. Gradient Ascent-Descent

For $f(\cdot, \mathbf{y})$ convex and $f(\mathbf{x}, \cdot)$ concave (with at least one of them strict), consider the coupled ODEs

$$\begin{aligned}\dot{\mathbf{x}}(t) &= -\nabla_{\mathbf{x}} f\left(\mathbf{x}(t), \mathbf{y}(t)\right)\\ \dot{\mathbf{y}}(t) &= \nabla_{\mathbf{y}} f\left(\mathbf{x}(t), \mathbf{y}(t)\right).\end{aligned}$$

Here, we take $V(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{x}^*\|^2 + \|\mathbf{y} - \mathbf{y}^*\|^2$ where $(\mathbf{x}^*, \mathbf{y}^*)$ is a saddle point. Then,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}V(\mathbf{x}, \mathbf{y}) &= -\left\langle \mathbf{x}(t) - \mathbf{x}^*, \nabla_{\mathbf{x}} f(\mathbf{x}(t), \mathbf{y}(t))\right\rangle + \left\langle \mathbf{y}(t) - \mathbf{y}^*, \nabla_{\mathbf{y}} f(\mathbf{x}(t), \mathbf{y}(t))\right\rangle\\ &\leq \left(f(\mathbf{x}*, \mathbf{y}(t)) - f(\mathbf{x}(t), \mathbf{y}(t))\right) + \left(f(\mathbf{x}(t), \mathbf{y}(t)) - f(\mathbf{x}(t), \mathbf{y}^*)\right)\\ &= \left(f(\mathbf{x}*, \mathbf{y}(t)) - f(\mathbf{x}(t), \mathbf{y}^*)\right) + \left(f(\mathbf{x}(t), \mathbf{y}^*) - f(\mathbf{x}(t), \mathbf{y}^*)\right)\\ &\leq 0\end{aligned}$$

with strict inequality away from the saddle point. Thus, this is a Liapunov function. This is particularly useful for primal-dual methods for the problem

$$\text{minimise } f(\mathbf{x}) \quad \text{subject to} \quad g(\mathbf{x}) \leq C.$$

Here, we do

$$\begin{aligned}\mathbf{x}_{n+1} &= \mathbf{x}_n - a(n)\left[\nabla f(\mathbf{x}_n) + \boldsymbol{\lambda}_n^\top \nabla g(\mathbf{x}_n) + M_{n+1}\right]\\ \boldsymbol{\lambda}_{n+1} &= \boldsymbol{\lambda}_n + a(n)\left[g(\mathbf{x}_n) - C\right].\end{aligned}$$

Here, we descend along $\mathbf{x}$ and ascend along $\boldsymbol{\lambda}$. We are seeking the saddle point of the Lagrangian, defined as

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) := f(\mathbf{x}) + \boldsymbol{\lambda}^\top (g(\mathbf{x}) - C).$$

Thus, we have

$$\begin{aligned}\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &= \nabla f(\mathbf{x}) + \boldsymbol{\lambda}^\top \nabla g(\mathbf{x})\\ \nabla_{\boldsymbol{\lambda}} \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) &= g(\mathbf{x}) - C.\end{aligned}$$

§§1.5.4. Fixed Point Schemes

Suppose $F\colon \mathbb{R}^d \to \mathbb{R}^d$ is Lipschitz. We wish to find a fixed point of F, that is, we wish to find $\mathbf{x}^*$ such that $F(\mathbf{x}^*) = \mathbf{x}^*$. Suppose F is a contraction map. That is, for one of

$$\|\mathbf{x}\|_p := \left(\sum_{i=1}^{d} |x_i|^p\right)^{\frac{1}{p}}, 1 \leq p < \infty \quad \|\mathbf{x}\|_\infty := \max_i |x_i|,$$

we have

$$\|F(\mathbf{x}) - F(\mathbf{y})\|_p \leq \alpha \|\mathbf{x} - \mathbf{y}\|_p \quad \forall \mathbf{x}, \mathbf{y},$$

where $0 < \alpha < 1$. Then, there is a unique fixed point $\mathbf{x}^*$ and $\mathbf{x}_n \to \mathbf{x}^*$ almost surely. In this case, $V(\mathbf{x}) := \|\mathbf{x} - \mathbf{x}^*\|_p$ works as a Liapunov function.

This also works

1. *Weighted norms.*

$$\|\mathbf{x}\|_{\mathbf{w},p} := \left(\sum_{i=1}^{d} w_i |x_i|^p \right)^{\frac{1}{p}}$$

2. *Pseudo-contractions.*

$$\|F(\mathbf{x}) - F(\mathbf{x}^*)\|_p \leq \alpha \|\mathbf{x} - \mathbf{x}^*\|_p \quad \forall \mathbf{x},$$

   where $\mathbf{x}^*$ is the fixed point, $0 < \alpha < 1$, and $1 < p \leq \infty$.

3. *Anti-monotone f.*

$$\langle F(\mathbf{x}) - F(\mathbf{y}), \mathbf{y} - \mathbf{x} \rangle < 0 \quad \text{for } \mathbf{x} \neq \mathbf{y}.$$

   The above inequality also generalises monotonicity to higher-dimensional and more general spaces.

4. *Special cases of 'non-expansive' maps.*

$$\|F(\mathbf{x}) - F(\mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y}.$$

§§1.5.5. Replicator Dynamics

This is an application in the field of evolutionary biology and evolutionary game theory where the iterates are in the probability simplex. We have

$$\dot{p}_i(t) = p_i(t) f_i(\mathbf{p}(t)) - \sum_j p_j(t) f_j(\mathbf{p}(t)) \tag{$\star$}$$

Here, $p_i(t)$ is the fraction of species i at time t and $f_i(\mathbf{p})$ denotes the payoff to species i if the current population profile is $\mathbf{p}$. Thus, if the payoff for species i is higher than average, then the fraction of species i increases. The probability simplex

$$\mathcal{S} := \left\{ \mathbf{x} = [x_1, \ldots, x_d] \in \mathbb{R}^d \colon x_i \geq 0 \, \forall i \, \sum_i x_i = 1 \right\}$$

is invariant under $(\star)$, as are its faces corresponding to one or more $x_i = 0$. The stable equilibria here correspond to 'evolutionary stable equilibria', that is, equilibria that are stable under single mutations.

The above iterates converge, for example, if $f_i = \frac{\partial F}{\partial x_i}$, in which case this is called a 'potential game' with potential F. In this case, we take $V = -F$. We have

$$\frac{\mathrm{d}}{\mathrm{d}t}F(\mathbf{x}(t)) = \sum_i p_i(t)\left(\frac{\partial F}{\partial x_i}(\mathbf{p}(t))\right)^2 - \left(\sum_i p_i(t)\frac{\partial F}{\partial x_i}(\mathbf{p}(t))\right)^2$$

This is positive unless ∇F is a constant vector, in which case the gradient along the probability simplex $\mathcal{S}$ is zero.