

# DATA DRIVEN TECHNIQUES

---


## Lecture 5 Learning Algorithms for Equation Solving

---

---

---

---



# SOLUTION OF UNKNOWN EQUATIONS

①

In the previous lecture we have seen ways to solve a system of equations

$$\left. \begin{array}{l} g_1(\theta_1, \theta_2, \dots, \theta_L) = 0 \\ g_2(\theta_1, \theta_2, \dots, \theta_L) = 0 \\ \vdots \\ g_L(\theta_1, \theta_2, \dots, \theta_L) = 0 \end{array} \right\} \rightarrow G(\theta) = 0$$

with a simple iterative algorithm

$$\theta_t = \theta_{t-1} + \mu G(\theta_{t-1}) \quad \mu \neq 0$$

where if  $\{\theta_t\}$  converges,  $\theta_t \rightarrow \theta_\infty$  then  $G(\theta_\infty) = 0$

We generalized to

$$\theta_t = \theta_{t-1} + \mu Q G(\theta_{t-1})$$

where  $Q$  invertible matrix  
and  $\mu \neq 0$

If  $\theta_t \rightarrow \theta_\infty$  then  $G(\theta_\infty) = 0$

If  $\theta_*$  is a root,  $G(\theta_*) = 0$  (2)

then  $\theta_*$  is possible to compute with the following iteration

$$\theta_{t+1} = \theta_t + \mu Q G(\theta_t)$$

If we define  $\Omega(\theta) = \nabla_{\theta} G(\theta)$  the Jacobian of  $G(\theta)$

$$\Omega(\theta) = \begin{bmatrix} \frac{\partial g_1}{\partial \theta_1} & \frac{\partial g_1}{\partial \theta_2} & \dots & \frac{\partial g_1}{\partial \theta_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_L}{\partial \theta_1} & \frac{\partial g_L}{\partial \theta_2} & \dots & \frac{\partial g_L}{\partial \theta_n} \end{bmatrix} \quad \text{and if } Q\Omega(\theta_*)$$

has eigenvalues with negative real part

then for sufficiently small  $\mu > 0$ , the iteration converges to  $\theta_*$  provided we start close enough to  $\theta_*$

$\theta_*$  is a locally stable equilibrium point of the iteration.

Can I solve  $G(\theta) = 0$  without knowing  $G(\theta)$ ? ③

What do I know instead of  $G(\theta)$ ? We are given a function  $H(x, \theta)$

$$H(x, \theta) = \begin{bmatrix} h_1(x_1, \dots, x_K, \theta_1, \theta_2, \dots, \theta_L) \\ \vdots \\ h_L(x_1, \dots, x_K, \theta_1, \dots, \theta_L) \end{bmatrix}$$

$H(x, \theta)$  and  $G(\theta)$  are related as follows: I have a random vector  $\mathcal{X}$  of dimension  $K$  and we relate the two functions as:

$$G(\theta) = \mathbb{E}_{\mathcal{X}}[H(\mathcal{X}, \theta)]$$

Although  $H(x, \theta)$  is known and constant,  $G(\theta)$  changes since it depends on the statistic of  $\mathcal{X}$ .

$$G(\theta) = \int H(x, \theta) f_{\mathcal{X}}(x) dx$$



If density changes from one problem to the other then  $G(\theta)$  is different and so are the roots of  $G(\theta)$ . ④

If  $f_X(x)$  is known then there is no difference with classical problem of deterministic and known  $G(\theta)$ .

Suppose  $f_X(x)$  **UNKNOWN**. Instead assume availability of realizations  $X_1, X_2, \dots, X_n$  of the random vector  $X$ .

Possibility to estimate roots of  $G(\theta) = \mathbb{E}[H(X, \theta)] = 0$

↳ by using the LLN

$$G(\theta) = \mathbb{E}[H(X, \theta)] \approx \frac{1}{n} \sum_{i=1}^n H(X_i, \theta) = \hat{G}(\theta)$$

↑  
Deterministic

Random, since it depends on random data.

I can now apply the simple iteration:

5

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \mu Q \hat{G}(\hat{\theta}_{t-1}) = \hat{\theta}_{t-1} + \mu Q \frac{1}{n} \sum_{i=1}^n H(x_i, \hat{\theta}_{t-1})$$

combine them to  $\frac{1}{n}$

Collection  $\{x_1, \dots, x_n\}$  is called training set.

We do not know  $G(\theta)$  but with the help of the training set we learn it and estimate the corresponding solution  $\theta_*$ .

Each iteration requires the computation of the vector function  $H(x, \theta)$  for ALL samples of the training set.

→ Big volume of computations if  $n$  is large.

### SIMPLIFIED ALGORITHM

$$\theta_t = \theta_{t-1} + \mu Q \cancel{\mathbb{E}_x}^{G(\theta_{t-1})} [H(x, \theta_{t-1})] \Rightarrow$$

replaced by  $\mathcal{D}$  data

$$\tilde{\theta}_t = \tilde{\theta}_{t-1} + \mu Q H(x_t, \tilde{\theta}_{t-1})$$

Where  $X_t$  runs over the samples of the training set. When samples are exhausted we restart from the beginning. (6)

The usage of all data is called cycle or epoch.

We compute one  $H(x, \theta)$  per iteration. Therefore for less complexity per iteration.

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \eta Q \frac{1}{n} \sum_{i=1}^n H(x_i, \hat{\theta}_{t-1})$$

The limit  $\hat{\theta}_t \rightarrow \hat{\theta}_\infty$  solution of the equation  $\hat{G}(\theta) = 0$ .  
The more data we have

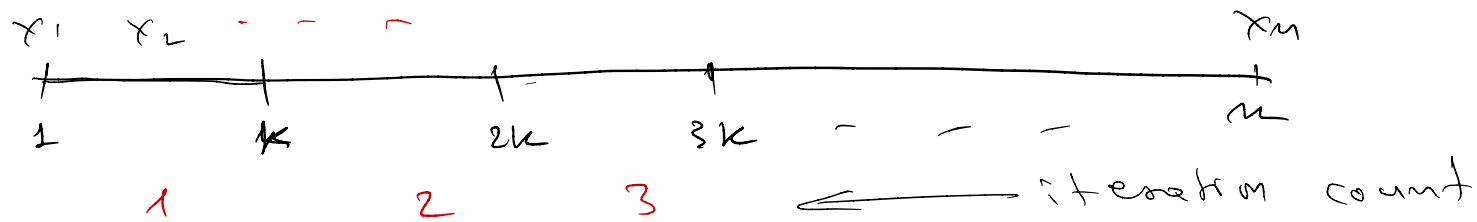
the better  $\hat{G}(\theta)$  approximates  $G(\theta)$  and therefore  $\hat{\theta}_\infty \rightarrow \theta_* : G(\theta_*) = 0$

$$\tilde{\theta}_t = \tilde{\theta}_{t+1} + \eta Q H(x_t, \tilde{\theta}_{t+1})$$

Where does  $\{\tilde{\theta}_t\}$  converge?  
Is  $\tilde{\theta}_\infty$  related to  $\theta_*$ ?

Will show that when  $\{\tilde{\theta}_t\}$  converges to  $\tilde{\theta}_\infty$ , this is also an approximation to  $\theta_*$

There are also intermediate versions instead of one or all data per iteration (8)



$$\checkmark \theta_t = \checkmark \theta_{t-1} + \frac{1}{K} \sum_{i=(t-1)k+1}^{tk} H(x_i, \checkmark \theta_{t-1})$$

Known as (micro)block algorithms

There are papers claiming that usage of (micro)blocks increases convergence speed.

Such claims are WRONG since we can prove that that blocks of size  $k > 1$  have EXACTLY the same convergence behavior as the classical version with  $k=1$ .

With blocks you can apply block-processing that could improve physical time computation. But iteration-wise there is no gain.

$$\check{\theta}_t = \check{\theta}_{t-1} + \mu \left\{ H(X_{(t-1)k+1}, \check{\theta}_{t-1}) + \dots + H(X_{(t-1)k+n}, \check{\theta}_{t-1}) \right\} \quad (9)$$

Micro-batches.

$K t_{mb} \rightarrow t$  relationship between micro-batch time and single-sample time

$$\tilde{\theta}_t = \tilde{\theta}_{t-1} + \mu H(X_t, \tilde{\theta}_{t-1}) \quad \text{classical algorithm}$$

Apply for data inside a batch

$$\check{\theta}_{tk} = \check{\theta}_{tk-1} + \mu H(X_{(t-1)k+n}, \check{\theta}_{tk-1})$$

$$\check{\theta}_{tk-1} = \check{\theta}_{tk-2} + \mu H(X_{(t-1)k+n-1}, \check{\theta}_{tk-2})$$

$\vdots$

$$\check{\theta}_{(t-1)k+1} = \check{\theta}_{(t-1)k} + \mu H(X_{(t-1)k+1}, \check{\theta}_{(t-1)k})$$

$$\left. \begin{array}{l} \check{\theta}_{(t-1)k+n-1} \\ \check{\theta}_{(t-1)k+n-2} \\ \vdots \\ \check{\theta}_{(t-1)k} \end{array} \right\}$$

We know

$$\check{\theta}_{(t-1)k+i} = \check{\theta}_{(t-1)k} + O(\mu)$$

$$\tilde{\Theta}_{(t-1)n+n} = \tilde{\Theta}_{(t-1)n} + \left\{ H(X_{(t-1)n+n}, \tilde{\Theta}_{(t-1)n}) + \dots + H(X_{(t-1)n+1}, \tilde{\Theta}_{(t-1)n}) \right\} \quad (15)$$

Batch with  $t \rightarrow kt$

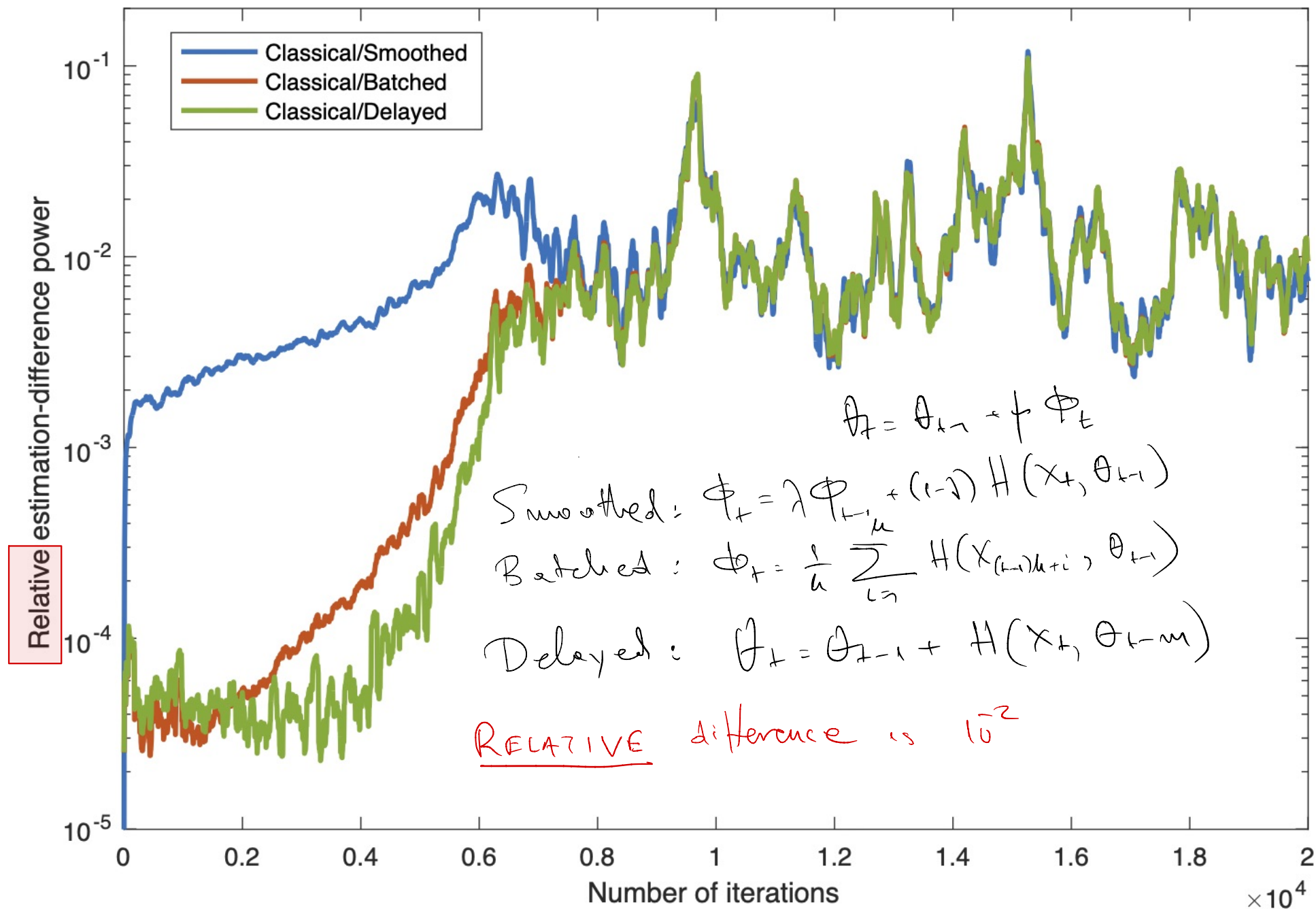
$$\check{\Theta}_{(t-1)n+n} = \check{\Theta}_{(t-1)n} + \left\{ H(X_{(t-1)n+n}, \check{\Theta}_{(t-1)n}) + \dots + H(X_{(t-1)n+1}, \check{\Theta}_{(t-1)n}) \right\}$$

But since  $\tilde{\Theta}_{(t-1)n+i} = \check{\Theta}_{(t-1)n} + O(\epsilon)$

$$\tilde{\Theta}_{(t-1)n+n} = \check{\Theta}_{(t-1)n} + \left\{ H(X_{(t-1)n+n}, \tilde{\Theta}_{(t-1)n}) + \dots + H(X_{(t-1)n+1}, \tilde{\Theta}_{(t-1)n}) \right\} + O(\epsilon^2)$$

differ by  $O(\epsilon^2)$  negligible term

negligible



## GRAD COMPUTATION

12

Consider the iterations

$$\hat{\theta}_t = \hat{\theta}_{t-1} + \frac{1}{n} \sum_{i=1}^n H(x_i, \hat{\theta}_{t-1})$$

We recall that  $G(\theta) = \mathbb{E}_x[H(x, \theta)]$  and

$$G(\theta_*) = 0.$$

Iteration converges to  $\hat{\theta}_\infty$  which satisfies  $\frac{1}{n} \sum_{i=1}^n H(x_i, \hat{\theta}_\infty) = 0$

Assume  $\hat{\theta}_\infty$  close to  $\theta_*$  and write  $\varepsilon = \hat{\theta}_\infty - \theta_*$

$$\sum_{i=1}^n H(x_i, \theta_* + \varepsilon) = 0 \rightarrow \sum_{i=1}^n H(x_i, \theta_*) + \sum_{i=1}^n \Omega(x_i, \theta_*) \varepsilon \approx 0$$

$\Omega(x, \theta)$  is Jacobian with respect to  $\theta$  of  $H(x, \theta)$



$$\sqrt{n} \mathcal{E} = - \left( \frac{1}{n} \sum_{i=1}^n \Omega(x_i, \theta_*) \right)^T \left( \frac{1}{\sqrt{n}} \sum_{i=1}^n H(x_i, \theta_*) \right) \text{ multiplied by } \sqrt{n} \quad (13)$$

From the LLN  $\frac{1}{n} \sum_{i=1}^n \Omega(x_i, \theta_*) \approx \mathbb{E}[\Omega(x, \theta_*)]$

$$= \mathbb{E}_x \left[ \mathcal{J}_\theta H(x, \theta) \right] \Big|_{\theta=\theta_*} = \mathcal{J}_\theta \mathbb{E}_x [H(x, \theta)] \Big|_{\theta=\theta_*} = \mathcal{J}_\theta G(\theta) \Big|_{\theta_*} = A$$

Because  $\mathbb{E}_x [H(x, \theta_*)] = G(\theta_*) = 0$ . We have from CLT

$$\frac{1}{\sqrt{n}} \sum_{i=1}^n H(x_i, \theta_*) \sim \mathcal{N}(0, \Sigma_x)$$

$$\Sigma_x = \mathbb{E}_x [H(x, \theta_*) H^T(x, \theta_*)] \quad \text{When } \{x_m\} \text{ are iid}$$

$$\mathcal{J}_\theta \quad \sqrt{n} \mathcal{E} \sim -\bar{A}^T Z \quad Z \sim \mathcal{N}(0, \Sigma)$$

$$\text{This means that } \sqrt{n} \mathcal{E} \sim \mathcal{N}(0, \bar{A}^T \Sigma \bar{A})$$

The error is Gaussian zero mean Covariance matrix  $\bar{A}^T \Sigma \bar{A}$