

Periodic Step-Size Adaptation for Single-Pass On-Line Learning

Chun-Nan Hsu^{1,2}, Yu-Ming Chang¹, Han-Shen Huang¹ and Yuh-Jye Lee³

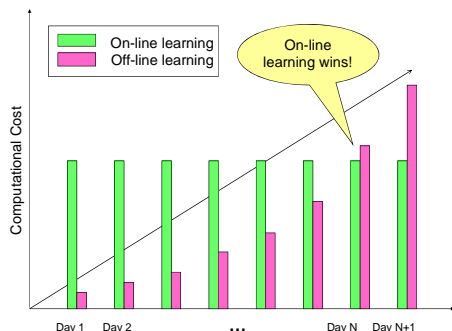
¹Institute of Information Science, Academia Sinica, Taipei, 115, Taiwan, ²USC/Information Science Institute, Marina del Rey, CA 90292, USA,

³Department of CSIE, National Taiwan University of Science and Technology.

chunnan@isi.edu, http://aiia.iis.sinica.edu.tw

1160

Single-pass online learning



Single-pass learning by 2SGD

- 2nd order SGD $\eta_*^{(t)} = \frac{H^{-1}(\theta^*; D)}{t+1}$
- **Good News** 2SGD can achieve empirical optimal in a single-pass (Bottou & LeCun 2004)

● **Bad News** computing H^{-1} is prohibitively expensive!!

Stability consideration

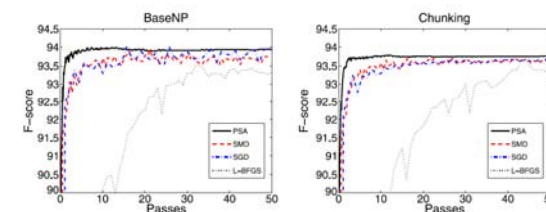
- SGD is a **stochastic** mapping. To reduce variance, let

$$M^b := \underbrace{M(M(\dots M(\theta)\dots))}_b$$

- We can estimate $\text{eig}(\mathbf{J}^b)$ by

$$\tilde{\gamma}_i^b = \frac{\theta_i^{(t+2b)} - \theta_i^{(t+b)}}{\theta_i^{(t+b)} - \theta_i^{(t)}}$$

Method (pass)	LS FD		LS OCR	
	accuracy	time	accuracy	time
Liblinear converge	96.74	4648.49	76.06	4454.42
Liblinear (1)	91.43	290.58	74.33	398.00
SvmSgd (20)	93.78	1135.67	-	-
SvmSgd (10)	93.77	567.68	73.71	473.35
SvmSgd (1)	93.60	56.78	73.76	46.96
PSA (1)	95.10	30.65	75.68	25.33



Why single-pass online learning?

- Incremental learning when training examples become available continuously
- Need to actually discard used training examples
- Loading data from disk to memory takes much longer than learning
- Simulate natural learners

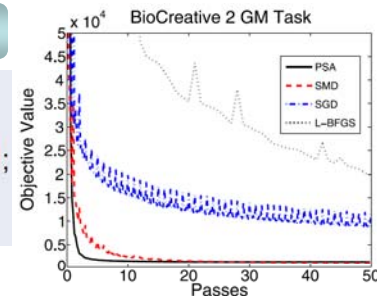
Approximating Jacobian

- Learning algorithms are fixed-point iteration mapping $\theta = M(\theta)$
- Taylor expansion gives $\theta^{(t+1)} = M(\theta^{(t)}) \approx \theta^* + J(\theta^{(t)} - \theta^*)$
- $\text{eig}(\mathbf{J})$ can be approximated by (Aitken 1925)

$$\gamma_i^{(t)} = \frac{\theta_i^{(t+2)} - \theta_i^{(t+1)}}{\theta_i^{(t+1)} - \theta_i^{(t)}}$$

Periodic step-size adaptation

- 1: **repeat** (from $\Theta^{(t)}$)
- 2: Obtain $\Theta^{(t+1)}, \dots, \Theta^{(t+2b)}$ by SGD with a fixed η ;
- 3: Estimate eigenvalues using $\Theta^{(t)}, \Theta^{(t+b)}$, and $\Theta^{(t+2b)}$;
- 4: Update η
- 5: **until** Convergence



On-line learning by SGD

Loss function given training examples

$$\mathcal{L}(\Theta; D) = \sum_{i=1}^{|D|} \mathcal{L}(\Theta; d_i) \approx \sum_{t=0}^{\frac{|D|}{b}-1} \mathcal{L}(\Theta; B^{(t)})$$

where $B^{(t)}$ is a batch of b examples $\subseteq D$.

Approximating H^{-1}

- Consider SGD mapping as a fixed-point iteration, too. $M(\theta) = \theta^{(t)} - \eta \nabla L(\theta; B^{(t)})$
- since $\mathbf{J} = \mathbf{M}' = \mathbf{I} - \eta \mathbf{H}$, we have $\text{eig}(\mathbf{J}) = \text{eig}(\mathbf{M}') = \text{eig}(\mathbf{I} - \eta \mathbf{H}) = 1 - \eta \text{eig}(\mathbf{H})$,

$$\text{eig}(H^{-1}) = \frac{\eta_i}{1 - \text{eig}(J)} \approx \frac{\eta_i}{1 - \gamma_i}$$

Now, we can approximate H^{-1} ! 😊

Experimental Results

Task	Model	Training	Test	Tag/Class	Weight	Target
Base NP	CRF	8936	2012	3	1015662	94.0% [10]
Chunking	CRF	8936	2012	23	7448606	93.6% [11]
BioNLP/NLPBA	CRF	18546	3856	11	5977675	70.0% [12]
BioCreative 2	CRF	15000	5000	3	10242972	86.5% [13]
LS FD	LSVM	2734900	2734900	2	900	3.26%
LS OCR	LSVM	1750000	1750000	2	1156	23.94%
MNIST [14]	CNN	60000	10000	10	134066	0.99%

Method (pass)	Base NP	Chunking	BioNLP/NLPBA	BioCreative 2
	time	F-score	time	F-score
SGD (1)	1.15	92.42	13.04	92.26
SMD (1)	41.50	91.81	350.00	91.89
PSA (1)	16.30	93.31	160.00	93.16
L-BFGS (batch)	221.17	93.91	8694.40	93.78

Method (pass)	time	error	mse	Method (pass)	time	error	mse
SGD (1)	266.77	2.36	2785.01	PSA w/o layer trick (1)	311.95	2.31	2389.65
SGD (140)	37336.20	0.99	82.22	PSA w/ layer trick (1)	311.00	1.97	2112.77
TONGA (n/a)	500.00	2.00	n/a	PSA re-start (1)	253.72	1.90	-

Summary

- PSA accurately approximates 2SGD to achieve near-optimal single-pass results for CRF, linear SVM and Convolutional neural nets
- PSA achieves this by periodically approximate $\text{eig}(H^{-1})$

References

- *Machine Learning Journal* version www.springerlink.com/index/b65483155lw83h20.pdf
- Source codes <http://aiia.iis.sinica.edu.tw>

Gradient Descent (off-line)	SGD (on-line)
1: Start from $\Theta^{(0)}$	1: Start from $\Theta^{(0)}$
2: repeat in iteration t	2: repeat in iteration t
3: $\Theta^{(t+1)} = \Theta^{(t)} - \eta \nabla \mathcal{L}(\Theta^{(t)}; D)$	3: $\Theta^{(t+1)} = \Theta^{(t)} - \eta \nabla \mathcal{L}(\Theta^{(t)}; B^{(t)})$
4: Update η	4: Update η
5: until Convergence	5: until Convergence