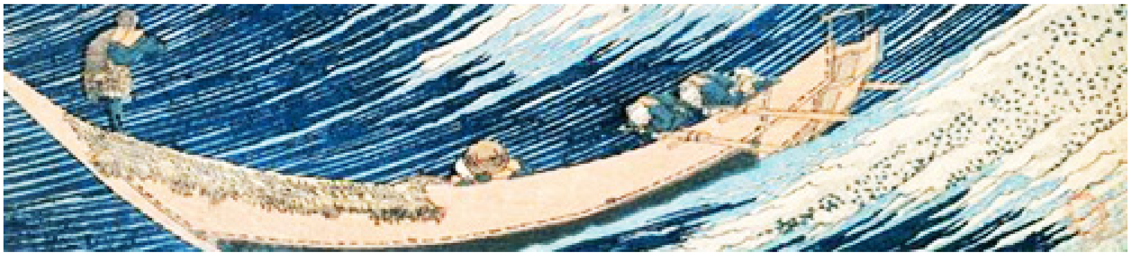
Mathematical Foundations of Data Sciences



Gabriel Peyré
CNRS & DMA
École Normale Supérieure
gabriel.peyre@ens.fr
www.gpeyre.com
www.numerical-tours.com

November 15, 2017

Chapter 17

Deep Learning

Before detailing deep architecture and their use, we start this chapter by presenting two essential computational tools that are used to train these model: stochastic optimization methods and automatic differentiation. In practice, they work hand-in-hand to be able to learn painlessly complicated non-linear model on large-scale datasets.

17.1 Stochastic Optimization

We detail stochastic Gradient Descent, with an application to the binary logistic classification problem.

We set the classes indexes to be $\{-1, +1\}$, and remove empty features, normalize X. n is the number of samples, p is the dimensionality of the features,

17.1.1 Batch Gradient Descent (BGD)

We first recall the usual deterministic (batch) gradient descent (BGD) on the problem of supervised logistic classification.

We refer to the dedicated section on logistic classification for background and more details about the derivations of the energy and its gradient.

Logistic classification aims at solving the following convex program

$$\min_w E(w) \overset{\text{def.}}{=} \frac{1}{n} \sum_{i=1}^n L(\langle x_i,\, w\rangle, y_i)$$

where the logistic loss reads

$$L(s, y) \overset{\text{def.}}{=} \log(1 + \exp(-sy))$$

We can define energy E and its gradient ∇E and use a vanilla gradient descent

$$w_{\ell+1} = w_\ell - \tau_\ell \nabla E(w_\ell).$$

17.1.2 Stochastic Gradient Descent (SGD)

As any empirical risk minimization procedure, the logistic classification minimization problem can be written as

$$\min_w E(w) = \frac{1}{n} \sum_i E_i(w) \quad \text{where} \quad E_i(w) = L(\langle x_i,\, w\rangle, y_i).$$

For very large n (which could in theory even be infinite, in which case the sum needs to be replaced by an expectation or equivalenty an integral), computing ∇E is prohebitive. It is possible instead to use a stochastic gradient descent (SGD) scheme

$$w_{\ell+1} = w_\ell - \tau_\ell \nabla E_{i(\ell)}(w_\ell)$$

where, for each iteration index ℓ, $i(\ell)$ is drawn uniformly at random in $\{1, \ldots, n\}$. Note that here

$$\nabla E_i(w) = x_i \nabla L(\langle x_i,\, w\rangle, y_i) \quad \text{where} \quad \nabla L(u, v) = v \odot \theta(-u)$$

Note that each step of a batch gradient descent has complexity $O(np)$, while a step of SGD only has complexity $O(p)$. SGD is thus advantageous when n is very large, and one cannot afford to do several passes through the data. In some situation, SGD can provide accurate results even with $\ell \ll n$, exploiting redundancy between the samples.

A crucial question is the choice of step size schedule τ_ℓ. It must tends to 0 in order to cancel the noise induced on the gradient by the stochastic sampling. But it should not go too fast to zero in order for the method to keep converging.

A typical schedule that ensures both properties is to have asymptically $\tau_\ell \sim \ell^{-1}$ for $\ell \to +\infty$. We thus propose to use

$$\tau_\ell \overset{\text{def.}}{=} \frac{\tau_0}{1 + \ell/\ell_0}$$

where ℓ_0 indicates roughly the number of iterations serving as a "warmup" phase.

One can prove the following convergence result

$$\mathbb{E}(E(w_\ell)) - E(w^\star) = O\left(\frac{1}{\sqrt{\ell}}\right),$$

where $\mathbb{E}$ indicates an expectation with respect to the i.i.d. sampling performed at each iteration.

We can perform the Stochastic gradient descent and display the evolution of the energy $E(w_\ell)$. One can overlay on top (black dashed curve) the convergence of the batch gradient descent, with a carefull scaling of the number of iteration to account for the fact that the complexity of a batch iteration is n times larger.

17.1.3 Stochastic Gradient Descent with Averaging (SGA)

Stochastic gradient descent is slow because of the fast decay of τ_ℓ toward zero.

To improve somehow the convergence speed, it is possible to average the past iterate, i.e. run a "classical" SGD on auxiliary variables $(\tilde{w}_\ell)_\ell$

$$\tilde{w}_{\ell+1} = \tilde{w}_\ell - \tau_\ell \nabla E_{i(\ell)}(\tilde{w}_\ell)$$

and output as estimated weight vector the average

$$w_\ell \overset{\text{def.}}{=} \frac{1}{\ell} \sum_{k=1}^{\ell} \tilde{w}_\ell.$$

This defines the Stochastic Gradient Descent with Averaging (SGA) algorithm.

Note that it is possible to avoid explicitly storing all the iterates by simply updating a running average as follow

$$w_{\ell+1} = \frac{1}{\ell}\tilde{w}_\ell + \frac{\ell - 1}{\ell} w_\ell.$$

In this case, a typical choice of decay is rather of the form

$$\tau_\ell \overset{\text{def.}}{=} \frac{\tau_0}{1 + \sqrt{\ell/\ell_0}}.$$

Notice that the step size now goes much slower to 0, at rate $\ell^{-1/2}$.

Typically, because the averaging stabilizes the iterates, the choice of (ℓ_0, τ_0) is less important than for SGD.

Bach proves that for logistic classification, it leads to a faster convergence (the constant involved are smaller) than SGD, since on contrast to SGD, SGA is adaptive to the local strong convexity of E.

We can display the evolution of the energy $E(w_\ell)$.

17.1.4 Stochastic Averaged Gradient Descent (SAG)

For problem size n where the dataset (of size $n \times p$) can fully fit into memory, it is possible to further improve the SGA method by bookeeping the previous gradient. This gives rise to the Stochastic Averaged Gradient Descent (SAG) algorithm.

We stored all the previously computed gradient in $(G^i)_{i=1}^n$, which necessitate $n \times p$ memory. The iterates are defined by using a proxy g for the batch gradient, which is progressively enhanced during the iterates.

The algorithm reads

$$h \leftarrow \nabla E_{i(\ell)}(\tilde{w}_\ell),$$

$$g \leftarrow g - G^{i(\ell)} + h,$$

$$G^{i(\ell)} \leftarrow h,$$

$$w_{\ell+1} = w_\ell - \tau g.$$

Note that in contrast to SGD and SGA, this method uses a fixed step size τ. Similarely to the BGD, in order to ensure convergence, the step size τ should be of the order of $1/L$ where L is the Lipschitz constant of E.

This algorithm improves over SGA and BGD since it has a convergence rate of $O(1/\ell$. Furthermore, in the presence of strong convexity (for instance when X is injective for logistic classification), it has a linear convergence rate, i.e.

$$\mathbb{E}(E(w_\ell)) - E(w^\star) = O\left(\rho^\ell\right),$$

for some $0 < \rho < 1$.

Note that this improvement over SGD and SGA is made possible only because SAG explicitly use the fact that n is finite (while SGD and SGA can be extended to infinite n and more general minimization of expectations).

We display the evolution of the energy $E(w_\ell)$.

17.2 Automatic Differentiation

17.3 Deep Discriminative Models

17.4 Deep Generative Models

17.4.1 Density Fitting

Fitting and MLE

Generative Models

17.4.2 Auto-encoders

17.4.3 GANs

Bibliography

[1] P. Alliez and C. Gotsman. Recent advances in compression of 3d meshes. In N. A. Dodgson, M. S. Floater, and M. A. Sabin, editors, *Advances in multiresolution for geometric modelling*, pages 3–26. Springer Verlag, 2005.

[2] P. Alliez, G. Ucelli, C. Gotsman, and M. Attene. Recent advances in remeshing of surfaces. In *AIM@SHAPE repport*. 2005.

[3] Amir Beck. *Introduction to Nonlinear Optimization: Theory, Algorithms, and Applications with MATLAB*. SIAM, 2014.

[4] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[5] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[6] E. Candès and D. Donoho. New tight frames of curvelets and optimal representations of objects with piecewise C^2 singularities. *Commun. on Pure and Appl. Math.*, 57(2):219–266, 2004.

[7] E. J. Candès. The restricted isometry property and its implications for compressed sensing. *Compte Rendus de l'Académie des Sciences*, Serie I(346):589–592, 2006.

[8] E. J. Candès, L. Demanet, D. L. Donoho, and L. Ying. Fast discrete curvelet transforms. *SIAM Multiscale Modeling and Simulation*, 5:861–899, 2005.

[9] A. Chambolle. An algorithm for total variation minimization and applications. *J. Math. Imaging Vis.*, 20:89–97, 2004.

[10] Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. *Theoretical foundations and numerical methods for sparse recovery*, 9(263-340):227, 2010.

[11] Antonin Chambolle and Thomas Pock. An introduction to continuous optimization for imaging. *Acta Numerica*, 25:161–319, 2016.

[12] S.S. Chen, D.L. Donoho, and M.A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1999.

[13] F. R. K. Chung. Spectral graph theory. *Regional Conference Series in Mathematics, American Mathematical Society*, 92:1–212, 1997.

[14] Philippe G Ciarlet. Introduction à l'analyse numérique matricielle et à l'optimisation. 1982.

[15] P. L. Combettes and V. R. Wajs. Signal recovery by proximal forward-backward splitting. *SIAM Multiscale Modeling and Simulation*, 4(4), 2005.

[16] P. Schroeder et al. D. Zorin. Subdivision surfaces in character animation. In *Course notes at SIGGRAPH 2000*, July 2000.

[17] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. on Pure and Appl. Math.*, 57:1413–1541, 2004.

[18] I. Daubechies and W. Sweldens. Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 4(3):245–267, 1998.

[19] D. Donoho and I. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81:425–455, Dec 1994.

[20] Heinz Werner Engl, Martin Hanke, and Andreas Neubauer. *Regularization of inverse problems*, volume 375. Springer Science & Business Media, 1996.

[21] M. Figueiredo and R. Nowak. An EM Algorithm for Wavelet-Based Image Restoration. *IEEE Trans. Image Proc.*, 12(8):906–916, 2003.

[22] M. S. Floater and K. Hormann. Surface parameterization: a tutorial and survey. In N. A. Dodgson, M. S. Floater, and M. A. Sabin, editors, *Advances in multiresolution for geometric modelling*, pages 157–186. Springer Verlag, 2005.

[23] Simon Foucart and Holger Rauhut. *A mathematical introduction to compressive sensing*, volume 1. Birkhäuser Basel, 2013.

[24] I. Guskov, W. Sweldens, and P. Schröder. Multiresolution signal processing for meshes. In Alyn Rockwood, editor, *Proceedings of the Conference on Computer Graphics (Siggraph99)*, pages 325–334. ACM Press, August8–13 1999.

[25] A. Khodakovsky, P. Schröder, and W. Sweldens. Progressive geometry compression. In *Proceedings of the Computer Graphics Conference 2000 (SIGGRAPH-00)*, pages 271–278, New York, July 23–28 2000. ACMPress.

[26] L. Kobbelt. $\sqrt{3}$ subdivision. In Sheila Hoffmeyer, editor, *Proc. of SIGGRAPH'00*, pages 103–112, New York, July 23–28 2000. ACMPress.

[27] M. Lounsbery, T. D. DeRose, and J. Warren. Multiresolution analysis for surfaces of arbitrary topological type. *ACM Trans. Graph.*, 16(1):34–73, 1997.

[28] S. Mallat. *A Wavelet Tour of Signal Processing, 3rd edition.* Academic Press, San Diego, 2009.

[29] Stephane Mallat. *A wavelet tour of signal processing: the sparse way.* Academic press, 2008.

[30] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Commun. on Pure and Appl. Math.*, 42:577–685, 1989.

[31] Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239, 2014.

[32] Gabriel Peyré. *L'algèbre discrète de la transformée de Fourier.* Ellipses, 2004.

[33] Gabriel Peyré and Marco Cuturi. Computational optimal transport. 2017.

[34] J. Portilla, V. Strela, M.J. Wainwright, and Simoncelli E.P. Image denoising using scale mixtures of Gaussians in the wavelet domain. *IEEE Trans. Image Proc.*, 12(11):1338–1351, November 2003.

[35] E. Praun and H. Hoppe. Spherical parametrization and remeshing. *ACM Transactions on Graphics*, 22(3):340–349, July 2003.

[36] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Phys. D*, 60(1-4):259–268, 1992.

[37] Filippo Santambrogio. Optimal transport for applied mathematicians. *Birkäuser, NY*, 2015.

[38] Otmar Scherzer, Markus Grasmair, Harald Grossauer, Markus Haltmeier, Frank Lenzen, and L Sirovich. *Variational methods in imaging.* Springer, 2009.

[39] P. Schröder and W. Sweldens. Spherical Wavelets: Efficiently Representing Functions on the Sphere. In *Proc. of SIGGRAPH 95*, pages 161–172, 1995.

[40] P. Schröder and W. Sweldens. Spherical wavelets: Texture processing. In P. Hanrahan and W. Purgathofer, editors, *Rendering Techniques '95.* Springer Verlag, Wien, New York, August 1995.

[41] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948.

[42] A. Sheffer, E. Praun, and K. Rose. Mesh parameterization methods and their applications. *Found. Trends. Comput. Graph. Vis.*, 2(2):105–171, 2006.

[43] Jean-Luc Starck, Fionn Murtagh, and Jalal Fadili. *Sparse image and signal processing: Wavelets and related geometric multiscale analysis.* Cambridge university press, 2015.

[44] W. Sweldens. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Applied and Computation Harmonic Analysis*, 3(2):186–200, 1996.

[45] W. Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1997.