Machine Learning @FB.com

Joan Bruna-Estrach, Soumith Chintala, Brian Karrer, Yann LeCun, Huamin Li, Serkan Piantino, Arthur Szlam, Andrew Tulloch, and me, Mark Tygert (+ many others on other projects)  May 1, 2019
Two BIG data projects

- PCA at scale ("PCA" = "principal component analysis")
- Mathematical motivation for complex-valued convnets ("convnet" = "convolutional network")
- This is a self-centric view of machine learning at Facebook: there are many, many, many other BIG data projects.
PCA at scale
PCA

- PCA ≈ SVD ("SVD" = "singular value decomposition"), possibly after centering or otherwise normalizing the columns of the matrix being analyzed or decomposed.

- Essentially synonymous are proper orthogonal decomposition, empirical modal analysis, empirical orthogonal functions, Karhunen-Loeve transform, Hotelling transform, . . . .

- In the form of low-rank approximation, PCA is the most popular method for unsupervised learning and data mining (clustering is common, too), dimension reduction, and denoising.

- As such, PCA is a critical component of many pipelines involving machine learning and data analytics, such as hubs-and-authorities (HITS) or latent semantic indexing (LSI) for information retrieval, canonical correlation analysis (CCA) for data mining, etc.
Low-rank approximation as an SVD

\[ A_{m \times n} \approx U_{m \times k} \cdot \Sigma_{k \times k} \cdot (V_{n \times k})^* \]

- \( k \) is the rank of the approximation.
- The columns of \( U \) are orthonormal.
- The columns of \( V \) are orthonormal.
- \( \Sigma \) is diagonal and all its entries are nonnegative.
SVD at scale at Facebook

- SciPy (now standard in scikit-learn) for memory-resident dense or sparse matrices, accelerated with Intel MKL BLAS
- Apache Spark for distributed dense matrices
- Apache Giraph for distributed sparse matrices
- Wrappers for Lua/Torch and FBLearner Flow
- Open sourced with permissive licenses
- Exhaustive benchmarkings complementing tight theoretical analysis demonstrate the dramatic superiority of our methods for low-rank approximation.
The basic idea

- The range of an approximately low-rank matrix is effectively low-dimensional.
- To identify the range, apply the matrix to random vectors, with the number of random vectors being near the dimension of the range.
- Project the matrix being analyzed on the low-dimensional approximation to the range.
- (Credit for the basics goes to many people, some here today.)
Critical subtleties

- Noise on real data often causes the singular values we desire to suppress to decay slowly and pollute the approximation; instead of applying the matrix being analyzed to each random vector only once, apply it several times, enhancing the decay of the tail of singular values (this solution came from my time at UCLA, not Facebook).

- Distributed orthonormalization is tricky to perform efficiently and numerically stably.

- Reduced accuracy in certain components of the algorithms enhances performance while not degrading the final accuracy of the low-rank approximation.
Mathematical motivation for complex-valued convnets
How and why does deep learning work?

- Nearly all the biggest advances have come from convolutional networks (convnets), not other forms of deep neural networks.
- Convnets (at least in the complex-valued instantiation we like) compute “data-driven multiscale windowed power spectra,” “data-driven multiscale windowed absolute spectra,” “data-driven multiwavelet absolute values,” or more generally “data-driven nonlinear multiwavelet packets.”
- Therefore, the remarkably rich and rigorous body of analysis for wavelets directly applies.
What is a convnet?

A convnet implements the repeated application of the following composition of three operations, recursively applying the composition to an input vector of nonnegative real numbers:

- convolution with complex-valued vectors, followed by
- taking the absolute value (or its square) of every entry of the vectors resulting from the convolutions, followed by
- local averaging (and then a square root of every entry resulting if the absolute values were squared)
What is the relation to windowed Fourier transforms?

- Admittedly, the sliding windowed Fourier transform is not just convolutions — the phases differ.
- However, the absolute values of windowed Fourier transforms are the absolute values of convolutions.
What is the relation to absolute values of wavelet coeffs.?

- Wavelet (or multiwavelet) transforms are linear “filter banks,” recursively applying convolutions to low frequency channels.
- Admittedly, linear filter banks omit taking absolute values.
- However, the lowest-frequency convolution in a windowed Fourier transform typically utilizes a nonnegative kernel — this kernel is the window — and the convolution of vectors whose entries are all nonnegative produces another vector whose entries are all nonnegative.
- Taking absolute values of the entries of a vector whose entries are all nonnegative leaves those entries unchanged.
- Local averaging is also convolution with a nonnegative kernel.
Should we all be optimizing complex-valued convnets?

Yes.
Are we all optimizing complex-valued convnets?

Yes, but only in our heads (in our networks of biological neurons), and in certain relatively simple yet still state-of-the-art systems for automatic speech recognition. For an elementary exposition from the perspective of generative modeling with stochastic processes, see http://tygert.com/ccnet.pdf — vol. 28, no. 5, pp. 815–825 of *Neural Computation*. 