Scalable, Distributed Factor Analysis in Spark

Daoyuan Jia, Tiancheng Liu, Danielle Rager

12/6/2016

Overview

- Scientific Motivation
- Dimensionality Reduction Models:
  - PCA / sPCA / FA
- Allen Institute Dataset and Data Pre-processing
  - Distributed Implementation
- Distributed Factor Analysis (FA) Algorithm
- Conclusion

Scientific Motivation

One research question in neuroscience:

Neurons exhibit highly variable electrical responses, even for the same stimuli. Neuroscientists want to understand the structure of trial-to-trial variability in neural responses. Are there global effects in variability across neurons in the network?

nature International weekly journal of science

NATURE | NEWS

Brain-data gold mine could reveal how neurons compute

Allen Brain Observatory releases unprecedented survey of activity in the mouse visual cortex.

Helen Shen

13 July 2016

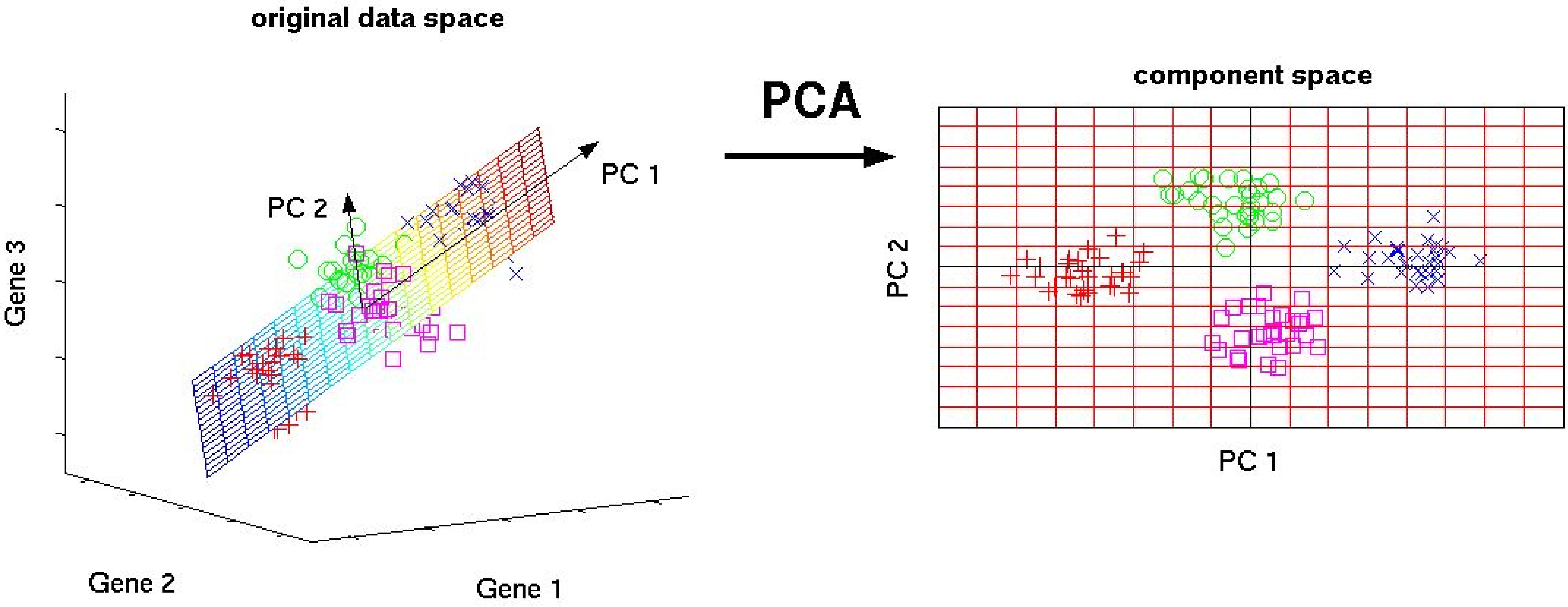
Noises Across Neurons in the Network

- Average human brain:100 billion neurons
- Proper dimensionality reduction techniques needed to analyze the variability across neurons
- Current algorithm: scalable
- Neuroscientists lack tools to analyze network covariability at this scale
- Solution: distributed dimensionality reduction algorithms

Overview

- Scientific Motivation
- Dimensionality Reduction Models:
  - PCA / sPCA / FA
- Allen Institute Dataset and Data Pre-processing
  - Distributed Implementation
- Distributed Factor Analysis (FA) Algorithm
- Conclusion

Dimensionality Reduction



1. Principal Component Analysis (PCA)

$$\underbrace{}_{}\overrightarrow{y} - \overrightarrow{\mu} = \overset{D \times d}{\overset{\downarrow}{C}}\overrightarrow{x}$$

- Closed-Form Solution: Singular Value Decomposition
- Complexity: $O(ND^2)$
- Distributed Implementation:

    Spark: MLlib-PCA; R: RScaLAPACK

2. Probabilistic Principal Component Analysis (PPCA)

$$\underbrace{\vec{y} - \vec{\mu} = \overbrace{C}^{D \times d}\vec{x} + \overbrace{\vec{\epsilon}}^{N(0, \sigma I)}}$$

- Expectation-Maximization (EM) Algorithm
- Complexity: $O(ND)$, better than PCA
- Distributed Implementation:

  Spark/Hadoop: Stochastic Principal Component Analysis (sPCA)

3. Factor Analysis (FA)

$$\vec{y} - \vec{\mu} = \overset{D \times d}{C}\vec{x} + \overset{N(0, \overset{D \times D}{\Psi})}{\vec{\epsilon}}$$

- Expectation-Maximization (EM) Algorithm
- Complexity: $O(ND)$
- Distributed Implementation:

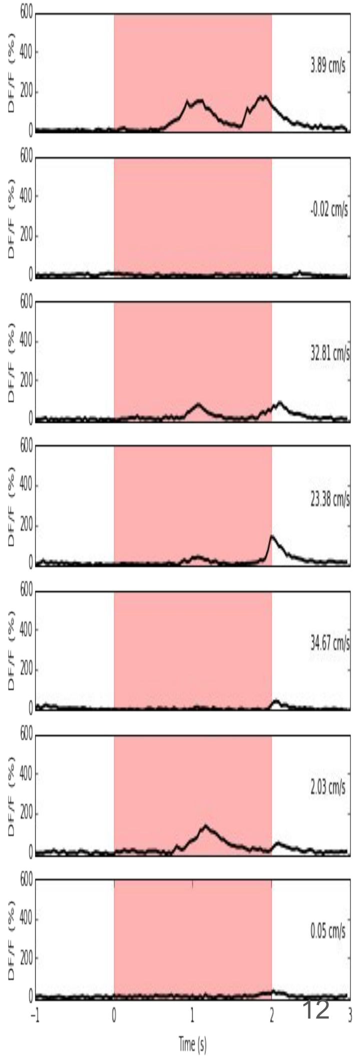
  Our implementation!

Overview

- Scientific Motivation
- Dimensionality Reduction Models:
  - PCA / sPCA / FA
- Allen Institute Dataset and Data Pre-processing
  - Distributed Implementation
- Distributed Factor Analysis (FA) Algorithm
- Conclusion

Allen Institute Dataset

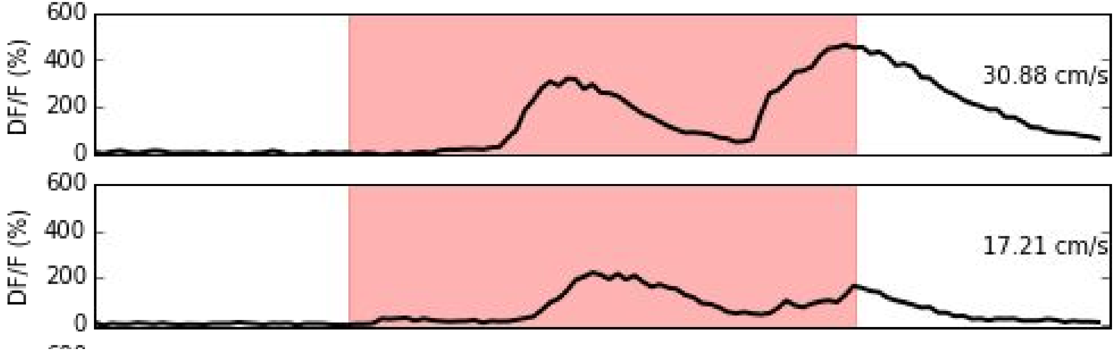
- AllenSDK:
  - For reading and processing data from Allen Institute
  - Contains an API to fetch data and experiment images
  - Also contains analytical functions for neuroscientists
- We focus on:
  - Raw data from Calcium Imaging
  - Cross-covariance between two cells’ calcium response
  - Final output: covariance matrix



Bottleneck: Calculating Covariance Matrix

Cell 1

Cell 2



- Between two cells: sum of cross-covariances up to lag *p*
- Complexity: $\mathrm{O}(p\mathrm{TN}^2) \approx \mathrm{O}(\mathrm{TN}^2)$
- On my laptop: 2hr for 1000 cells
- Solution: distributed matrix operations in Spark

Overview

- Scientific Motivation
- Dimensionality Reduction Models:
    - PCA / sPCA / FA
- Allen Institute Dataset and Data Pre-processing
    - Distributed Implementation
- Distributed Factor Analysis (FA) Algorithm
- Conclusion

EM algorithm for FA $\vec{y} - \vec{\mu} = C\vec{x} + \vec{\epsilon}$

$N(0, \Psi)$

- **E step:**
  - Fix C and Ψ
  - Compute conditional likelihood L(X|Y)

$$< X|Y_c >$$

$$< (X|Y_c)(X|Y_c)^T >$$

- **M step:**
  - Fix conditional likelihood
  - Compute new C and Ψ

EM Algorithm for FA

$$Y_c = Y - \mu$$

$$M = CC^T + \Psi$$

$$< X|Y_c >= X_m = C^T M^{-1} Y_c$$

$$< (X|Y_c)(X|Y_c)^T >= \Sigma_{X_m} = I - C^T M^{-1} C + X_m X_m^T$$

$$Y_{proj} = Y_c {X_m}^T$$

$$C_{new} = \Sigma_{X_m} Y_{proj}$$

$$\psi_{proj} = C_{new} X_m Y_m^T$$

$$\Psi_{new} = \frac{1}{N} \mathrm{diag}(Y_c {Y_c}^T - \psi_{proj})$$

sFA Optimizations: Distribute operations minimally

Driver program does most operations locally, launches only a few Spark jobs

distribute computations where you have dimension N

$$Y_c = Y - \mu$$

$$M = CC^T + \Psi$$

$$< X|Y_c >= X_m = C^T M^{-1} Y_c$$ **MapReduce**

$$< (X|Y_c)(X|Y_c)^T >= \Sigma_{X_m} = I - C^T M^{-1} C + X_m X_m^T$$ **MapReduce**

$$Y_{proj} = Y_c {X_m}^T$$ **MapReduce**

$$C_{new} = \Sigma_{X_m} Y_{proj}$$

$$\psi_{proj} = C_{new} X_m Y_m^T$$ **MapReduce**

$$\Psi_{new} = \frac{1}{N} \mathrm{diag}(Y_c {Y_c}^T - \psi_{proj})$$

sFA Optimizations: Distribute operations minimally

Use same MapReduce job for operations w/o dependencies

$$Y_c = Y - \mu$$

$$M = CC^T + \Psi$$

$$< X|Y_c >= X_m = C^T M^{-1} Y_c$$ **MapReduce**

$$< (X|Y_c)(X|Y_c)^T >= \Sigma_{X_m} = I - C^T M^{-1} C + X_m X_m^T$$

$$Y_{proj} = Y_c {X_m}^T$$

MapReduce

$$C_{new} = \Sigma_{X_m} Y_{proj}$$

$$\psi_{proj} = C_{new} X_m Y_m^T$$ **MapReduce**

$$\Psi_{new} = \frac{1}{N} \text{diag}(Y_c {Y_c}^T - \psi_{proj})$$

sFA Optimizations: Minimize Intermediary Data

Recompute X and Y at each job rather than storing and exchanging

$$Y_c = Y - \mu$$

$$M = CC^T + \Psi$$

$$< X|Y_c >= \boxed{X_m} = C^T M^{-1} Y_c$$

$$< (X|Y_c)(X|Y_c)^T >= \Sigma_{X_m} = I - C^T M^{-1} C + \boxed{X_m} X_m^T$$

$$Y_{proj} = Y_c \boxed{{X_m}^T}$$

$$C_{new} = \Sigma_{X_m} Y_{proj}$$

$$\psi_{proj} = C_{new} \boxed{X_m} Y_m^T$$

$$\Psi_{new} = \frac{1}{N} \mathrm{diag}(Y_c {Y_c}^T - \psi_{proj})$$

sFA Optimizations: Minimize Intermediary Data

Recompute X and Y at each job rather than storing and exchanging

$$Y_c = Y - \mu$$

$$M = CC^T + \Psi$$

$$\cancel{< X|Y_c > = X_m = C^T M^{-1} Y_c}$$

$$< (X|Y_c)(X|Y_c)^T > = \Sigma_{X_m} = I - C^T M^{-1} C + \boxed{X_m} X_m^T$$

$$Y_{proj} = Y_c {X_m}^T$$

MapReduce

$$C_{new} = \Sigma_{X_m} Y_{proj}$$

$$\psi_{proj} = C_{new} \boxed{X_m} Y_m^T$$

MapReduce

$$\Psi_{new} = \frac{1}{N} \mathrm{diag}(Y_c {Y_c}^T - \psi_{proj})$$

sFA Optimizations: Leverage Sparsity

(Don't lose opportunity to do computations on 0 values)

sparse

Mean sparsity: $M^{-1}Y_c = M^{-1}(Y - \mu) = M^{-1}\boxed{Y} - M^{-1}\mu$

Matrix Inversion Lemma:

$$M^{-1} = (CC^T + \Psi)^{-1}$$
$$= \boxed{\Psi^{-1}} - \boxed{\Psi^{-1}C}(I + C^T\Psi^{-1}C)^{-1}\boxed{C^T\Psi^{-1}}$$

Diagonal matrix multiplication tricks

sFA Optimizations: Efficient Matrix Multiplication

Optimizes computations when both matrices are large

$$(A * B)_i = A_i * B$$

Optimization when one matrix is small enough to fit in memory

Overview

- Scientific Motivation
- Dimensionality Reduction Models:
  - PCA / sPCA / FA
- Allen Institute Dataset and Data Pre-processing
  - Distributed Implementation
- Distributed Factor Analysis (FA) Algorithm
- Conclusion

Conclusion: Format for Large Distributed Files

H5Spark: Bridging the I/O Gap between Spark and Scientific Data Formats on HPC Systems

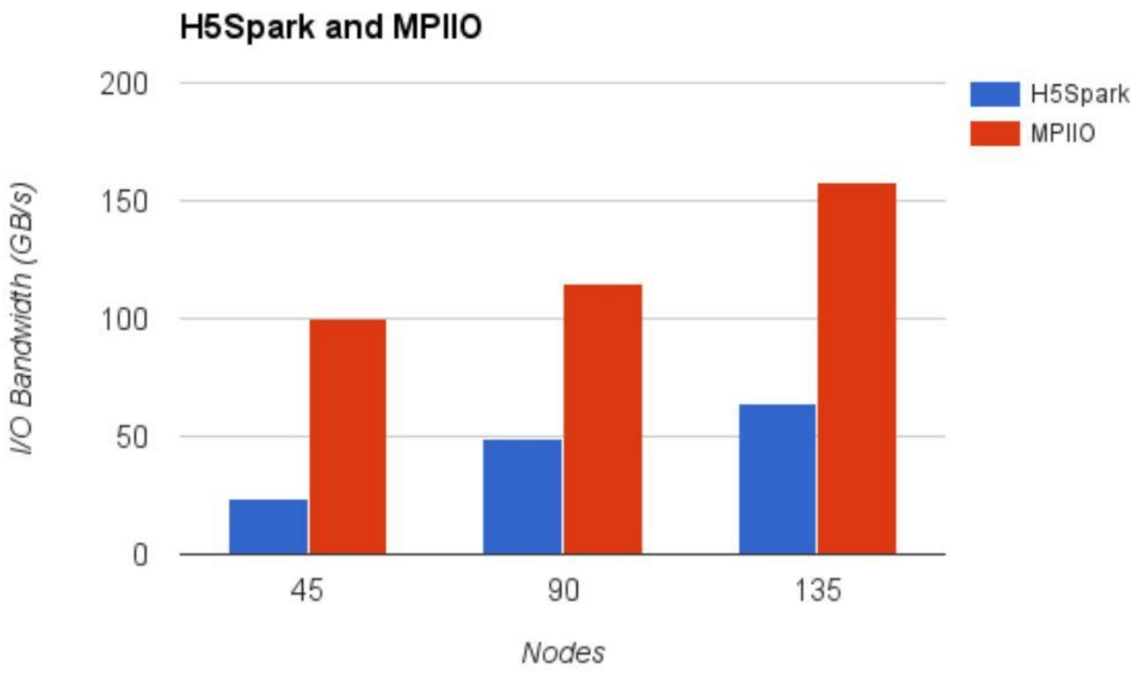
Jialin Liu[1], Evan Racah[1],Quincey Koziol[1], Richard Shane Canon[1], Alex Gittens[2], Lisa Gerhardt[1], Suren Byna[1], Mike F. Ringenburg[3], Prabhat[1].

- HDF5:
  - Hierarchical Data Format V
  - Flexible and efficient I/O
  - High volume and complex
  - NWB from Allen Institute
- HFSpark
  - Department of Energy
  - From HDF5 to Spark RDD
  - Implemented in JavaSpark
  - Superior performance

Conclusion: Format for Large Distributed Files

MPI IO

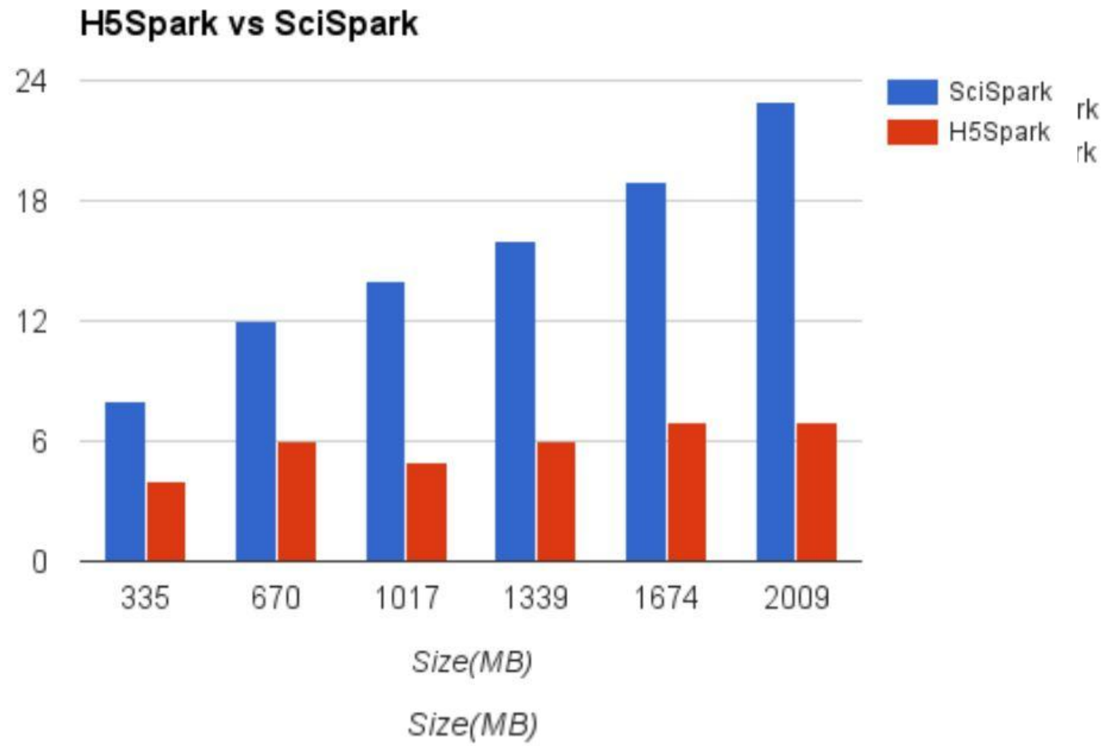
- IBM's Watson Laboratory
- Parallel I/O



Conclusion: Format for Large Distributed Files

SciSpark

- NASA
- scientific RDD (SRDD)



Reference

1. Allen Institute. “Brain Observatory Trace Analysis.” http://alleninstitute.github.io/AllenSDK/_static/examples/nb/brain_observatory_analysis.html#Drifting-Gratings
2. Tarek Elgamal, Maysam Yabandeh, Ashraf Aboulnaga, Waleed Mustafa, and Mohamed Hefeeda. 2015. sPCA: Scalable Principal Component Analysis for Big Data on Distributed Platforms. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (SIGMOD '15). ACM, New York, NY, USA, 79-91. DOI: http://dx.doi.org/10.1145/2723372.2751520
3. http://johnsonhsieh.github.io/DSR_workshop/image/fig_pca_principal_component_analysis.png
4. https://cug.org/proceedings/cug2016_proceedings/includes/files/pap137-file2.pdf
5. J.L. Liu, E. Racah, Q. Koziol, R. S. Canon, A. Gittens, L. Gerhardt, S. Byna, M. F. Ringenburg, Prabhat. "H5Spark: Bridging the I/O Gap between Spark and Scientific Data Formats on HPC Systems", Cray User Group, 2016