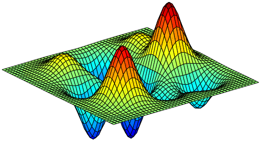
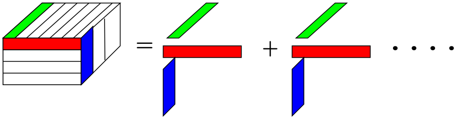
Guaranteed Non-convex Machine Learning Using Tensor Methods

Anima Anandkumar





U.C. Irvine

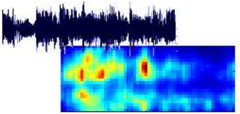
Regime of Modern Machine Learning

Massive datasets, growth in computation power, challenging tasks

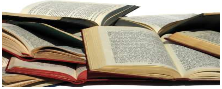
Success of Supervised Learning

- Learn $p(y|x)$ from labeled samples $\{(x_i, y_i)\}$.
- Extract relevant features from large amounts of labeled data.

Image classification



Speech recognition



Text processing

Regime of Modern Machine Learning

Massive datasets, growth in computation power, challenging tasks

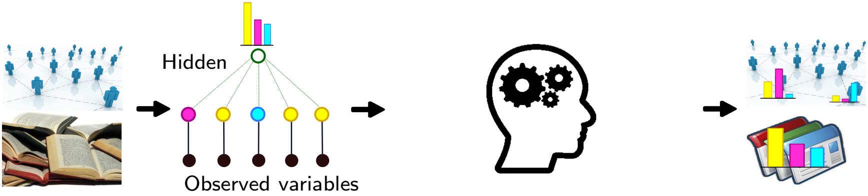
Missing Link in AI: Unsupervised Learning

- Learn $p(x)$ from unlabeled samples $\{x_i\}$.
- Discover latent variables related to observed variable x.
- Human vs. Machine Learning: Make discoveries automatically.

Image Credit: Whizz Education

Unsupervised Learning via Probabilistic Models

Data → Model → Learning Algorithm → Predictions



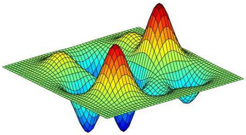
Challenges in High dimensional Learning

- Dimension of $x \gg$ dim. of latent variable h.
- Learning is like finding needle in a haystack.
- Computationally & statistically challenging.

Overview of Unsupervised Learning Methods

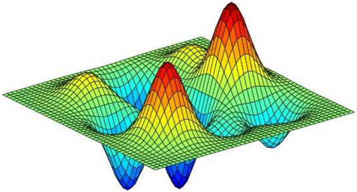
$\boxed{\text{Goal: learn model parameters } \theta \text{ from observations } x.}$

- Maximum likelihood: $\max_{\theta} p(x; \theta)$.
- Non-convex: stuck in local optima.
- Curse of dimensionality: Exponential no. of critical points.



- Heuristics: Expectation Maximization, Variational Inference ....
- Other mechanisms such as Generative Adversarial Nets also non-convex.

Guaranteed Learning through Tensor Methods



Replace the objective function

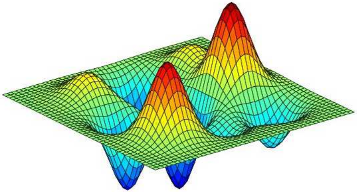
Max Likelihood vs. Best Tensor decomp.

Preserves Global Optimum (infinite samples)

$$\boxed{\arg\max_{\theta} p(x;\theta) = \arg\min_{\theta} \|\widehat{T}(x) - T(\theta)\|_{\mathbb{F}}^2}$$

$\widehat{T}(x)$: empirical tensor, $T(\theta)$: low rank tensor based on θ.

Guaranteed Learning through Tensor Methods



Replace the objective function

Max Likelihood vs. Best Tensor decomp.

Preserves Global Optimum (infinite samples)

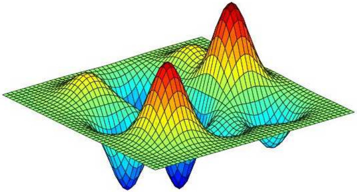
$$\boxed{\arg\max_{\theta} p(x;\theta) = \arg\min_{\theta} \|\widehat{T}(x) - T(\theta)\|_{\mathbb{F}}^2}$$

$\widehat{T}(x)$: empirical tensor, $T(\theta)$: low rank tensor based on θ.

Finding globally opt tensor decomposition

Simple algorithms succeed under mild and natural conditions for many learning problems.

Guaranteed Learning through Tensor Methods



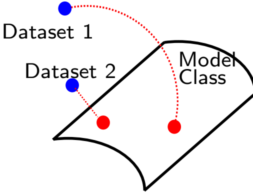
Replace the objective function

Max Likelihood vs. Best Tensor decomp.

Preserves Global Optimum (infinite samples)

$$\boxed{\arg\max_{\theta} p(x;\theta) = \arg\min_{\theta} \|\widehat{T}(x) - T(\theta)\|_{\mathbb{F}}^2}$$

$\widehat{T}(x)$: empirical tensor, $T(\theta)$: low rank tensor based on θ.

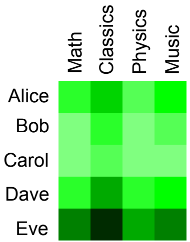


Finding globally opt tensor decomposition

Simple algorithms succeed under mild and natural conditions for many learning problems.

Outline

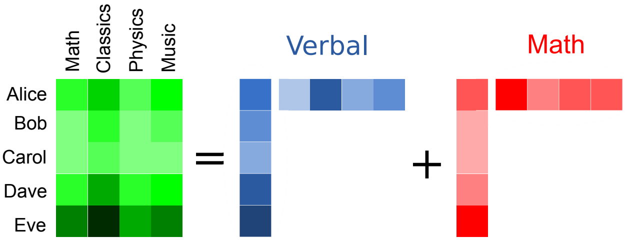
Matrix Decomposition: Discovering Latent Factors



- List of scores for students in different tests
- Learn hidden factors for Verbal and Mathematical Intelligence [C. Spearman 1904]

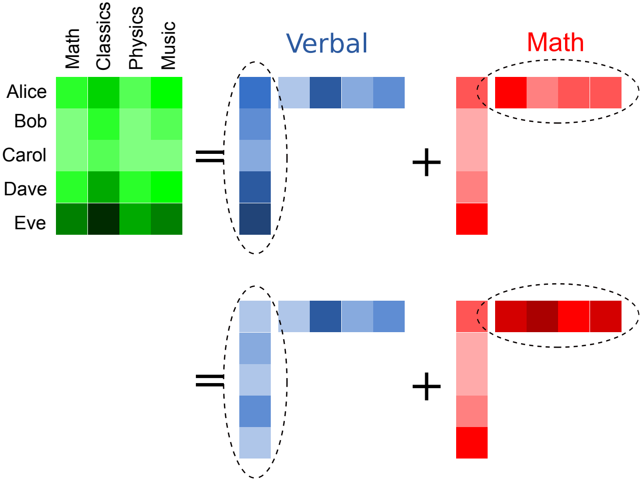
$$\text{Score (student,test)} = \text{student}_{\text{verbal-intlg}} \times \text{test}_{\text{verbal}} + \text{student}_{\text{math-intlg}} \times \text{test}_{\text{math}}$$

Matrix Decomposition: Discovering Latent Factors



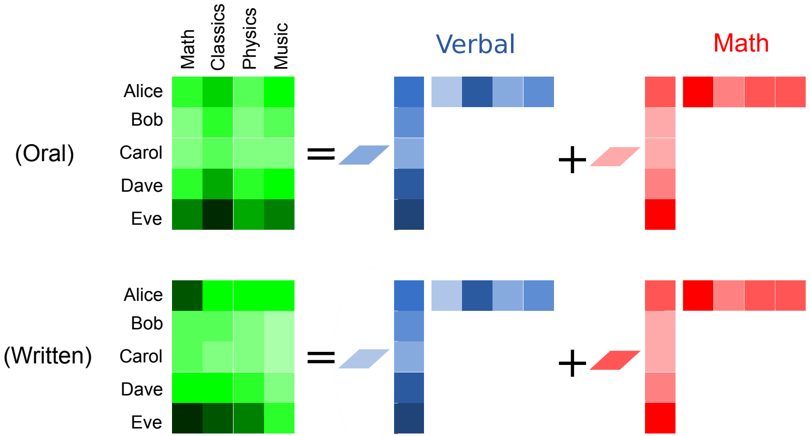
- Identifying hidden factors influencing the observations
- Characterized as matrix decomposition

Matrix Decomposition: Discovering Latent Factors



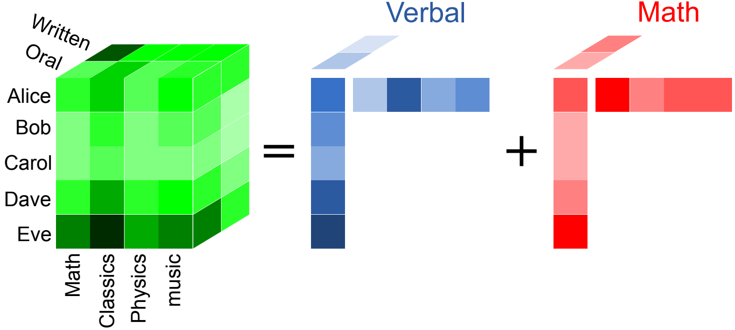
- Decomposition is not necessarily unique.
- Decomposition cannot be overcomplete.

Tensor: Shared Matrix Decomposition



- Shared decomposition with different scaling factors
- Combine matrix slices as a tensor

Tensor Decomposition



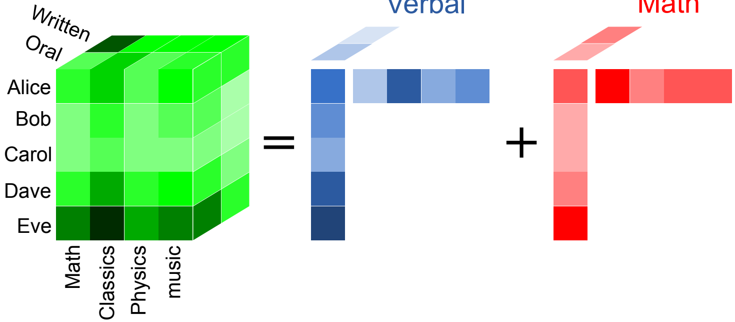
- Outer product notation:

$$T = u \otimes v \otimes w + \tilde{u} \otimes \tilde{v} \otimes \tilde{w}$$

$$\Updownarrow$$

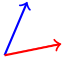
$$T_{i_1,i_2,i_3} = u_{i_1} \cdot v_{i_2} \cdot w_{i_3} + \tilde{u}_{i_1} \cdot \tilde{v}_{i_2} \cdot \tilde{w}_{i_3}$$

Tensor Decomposition

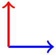


Uniqueness of Tensor Decomposition [J. Kruskal 1977]

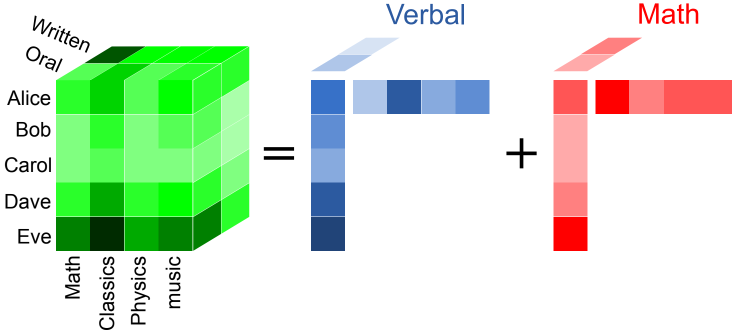
- Above tensor decomposition: unique when rank one pairs are linearly independent



- Matrix case: when rank one pairs are orthogonal



Tensor Decomposition



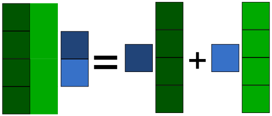
Finding Best Tensor Decomposition? Overcome Non-convexity?

Notion of Tensor Contraction

Extends the notion of matrix product

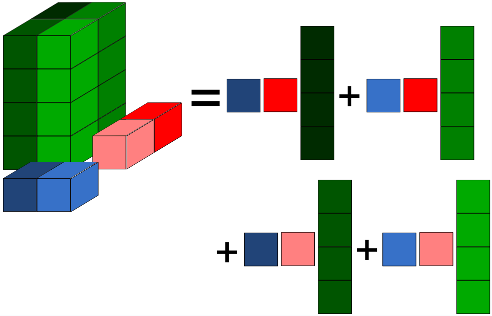
Matrix product

$$Mv = \sum_j v_j M_j$$

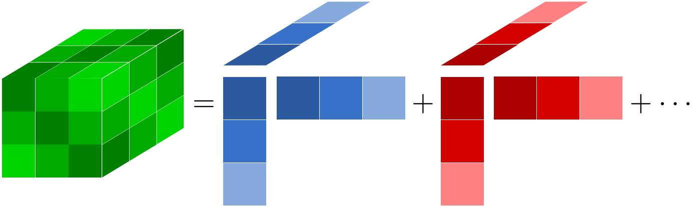


Tensor Contraction

$$T(u, v, \cdot) = \sum_{i,j} u_i v_j T_{i,j,:}$$



Symmetric Tensor Decomposition



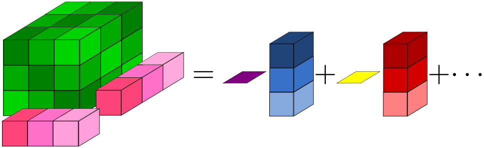
$$T = v_1^{\otimes 3} + v_2^{\otimes 3} + \cdots ,$$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

$$\boxed{v \mapsto \frac{T(v, v, \cdot)}{\|T(v, v, \cdot)\|}}.$$



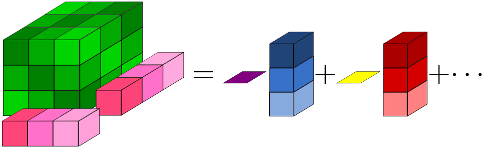
$$T(v, v, \cdot) = \langle v, v_1 \rangle^2 v_1 + \langle v, v_2 \rangle^2 v_2$$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

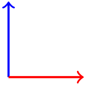
$$\boxed{v \mapsto \frac{T(v,v,\cdot)}{\|T(v,v,\cdot)\|}.}$$



$$T(v,v,\cdot) = \langle v, v_1\rangle^2 v_1 + \langle v, v_2\rangle^2 v_2$$

Orthogonal Tensors

- $\vec{v}_1 \perp \vec{v}_2$.
- $T(v_1, v_1, \cdot) = \lambda_1 v_1$.

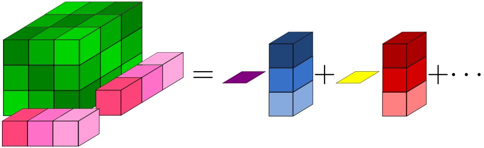


A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

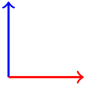
$$\boxed{v \mapsto \frac{T(v, v, \cdot)}{\|T(v, v, \cdot)\|}.}$$

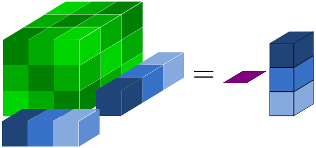


$$T(v, v, \cdot) = \langle v, v_1 \rangle^2 v_1 + \langle v, v_2 \rangle^2 v_2$$

Orthogonal Tensors

- $\color{blue}{\vec{v}_1} \perp \color{red}{\vec{v}_2}$.
- $T(\color{blue}{v_1}, \color{blue}{v_1}, \cdot) = \lambda_1 \color{blue}{v_1}$.



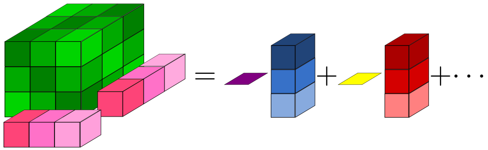


A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

$$\boxed{v \mapsto \frac{T(v,v,\cdot)}{\|T(v,v,\cdot)\|}.}$$



$$T(v,v,\cdot) = \langle v, v_1\rangle^2 v_1 + \langle v, v_2\rangle^2 v_2$$

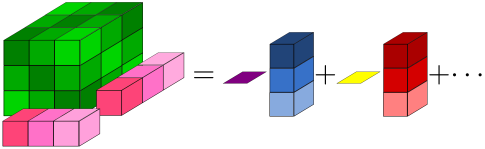
Exponential no. of stationary points for power method:
$T(v,v,\cdot) = \lambda v$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

$$\boxed{v \mapsto \frac{T(v, v, \cdot)}{\|T(v, v, \cdot)\|}.}$$

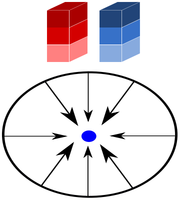


$$T(v, v, \cdot) = \langle v, v_1 \rangle^2 v_1 + \langle v, v_2 \rangle^2 v_2$$

Exponential no. of stationary points for power method:

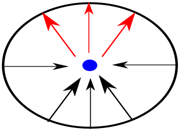
$T(v, v, \cdot) = \lambda v$

Stable



Unstable

Other statitionary points

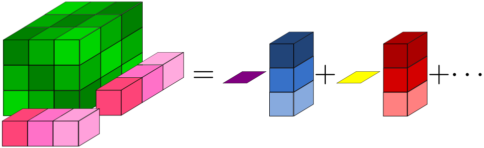


A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Symmetric Tensor Decomposition

Tensor Power Method

$$\boxed{v \mapsto \frac{T(v,v,\cdot)}{\|T(v,v,\cdot)\|}.}$$



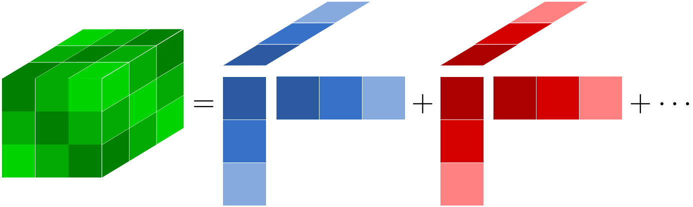
$$T(v,v,\cdot) = \langle v, v_1\rangle^2 v_1 + \langle v, v_2\rangle^2 v_2$$

Exponential no. of stationary points for power method:
$T(v,v,\cdot) = \lambda v$

For power method on orthogonal tensor, no spurious stable points

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

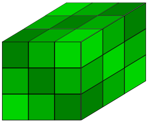


$$T = v_1^{\otimes 3} + v_2^{\otimes 3} + \cdots ,$$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

Orthogonalization

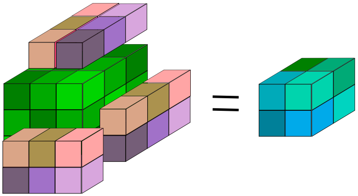


Input tensor T

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

Orthogonalization

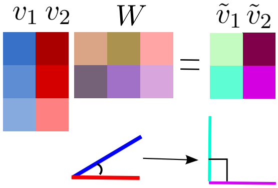


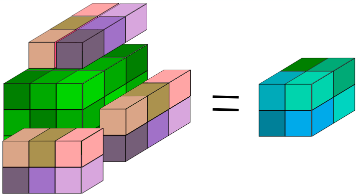
$$T(W, W, W) = \tilde{T}$$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

Orthogonalization



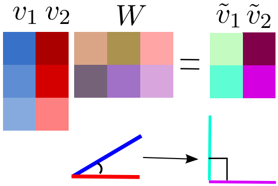


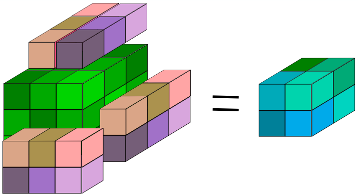
$$T(W, W, W) = \tilde{T}$$

A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

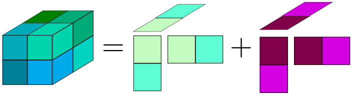
Orthogonalization





$$T(W,W,W) = \tilde{T}$$

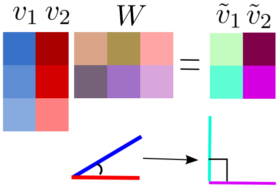
$$\tilde{T} = T(W,W,W) = \tilde{v_1}^{\otimes 3} + \tilde{v_2}^{\otimes 3} + \cdots,$$

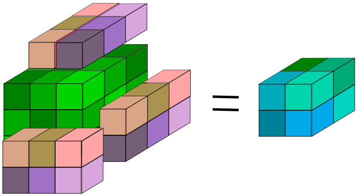


A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

Orthogonalization

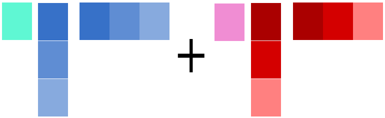


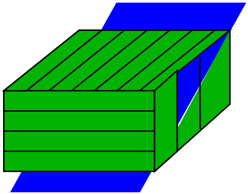


$$T(W, W, W) = \tilde{T}$$

Find W using SVD of Matrix Slice

$$M = T(\cdot, \cdot, \theta) =$$

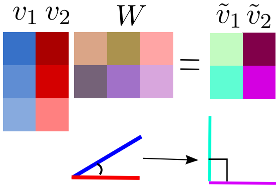


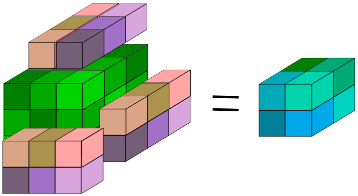


A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Non-orthogonal Tensor Decomposition

Orthogonalization





$$T(W, W, W) = \tilde{T}$$

Orthogonalization: invertible when v_i's linearly independent.

Guaranteed tensor decomposition: when v_i's linearly independent.

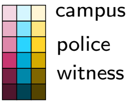
A., R. Ge, D. Hsu, S. Kakade, M. Telgarsky, "Tensor Decompositions for Learning Latent Variable Models," JMLR 2014.

Outline

Extracting Topics from Documents

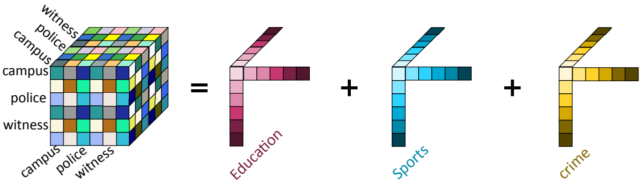
A., D. P. Foster, D. Hsu, S.M. Kakade, Y.K. Liu. "Two SVDs Suffice: Spectral decompositions for probabilistic topic modeling and latent Dirichlet allocation," NIPS 2012.

Tensor Methods for Topic Modeling

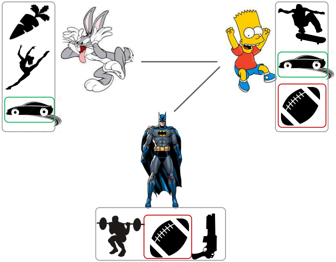


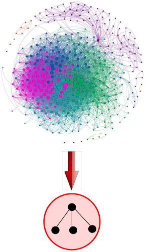
- Topic-word matrix $\boxed{\mathbb{P}[\text{word} = i | \text{topic} = j]}$
- Linearly independent columns

Moment Tensor: Co-occurrence of Word Triplets

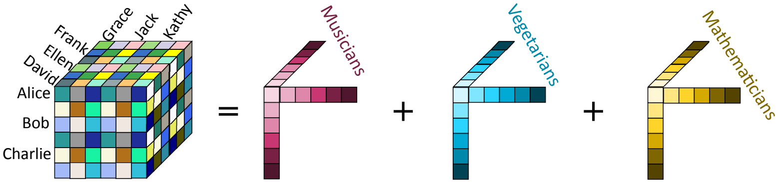


Extracting Communities in Social Networks





Moment Tensor: Common Friends among Node Triplets

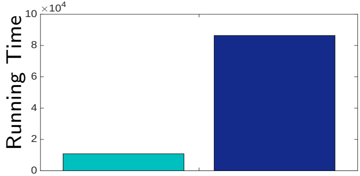


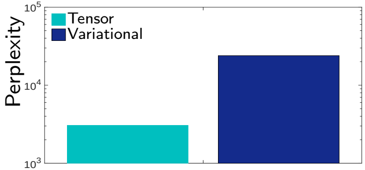
A., R. Ge, D. Hsu, S.M. Kakade. "A Tensor Spectral Approach to Learning Mixed Membership Community Models" COLT 2013.

Tensors vs. Variational Inference

Criterion: Perplexity $= \exp[-\text{likelihood}]$.

Learning Topics from PubMed on Spark, 8mil articles



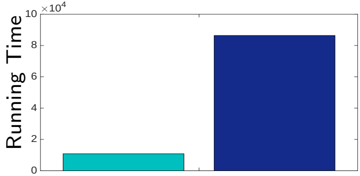


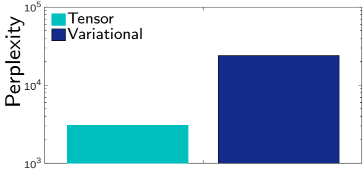
F. Huang, U.N. Niranjan, M. Hakeem, A, "Online tensor methods for training latent variable models," JMLR 2014.

Tensors vs. Variational Inference

Criterion: Perplexity $= \exp[-\text{likelihood}]$.

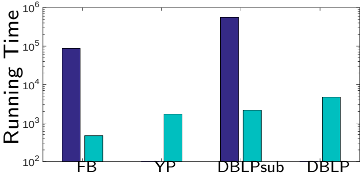
Learning Topics from PubMed on Spark, 8mil articles

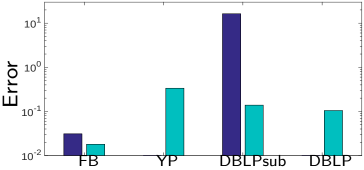




Learning network communities on single workstation

Facebook $n \sim 20k$, Yelp $n \sim 40k$, DBLP-sub $n \sim 1e5$, DBLP $n \sim 1e6$.



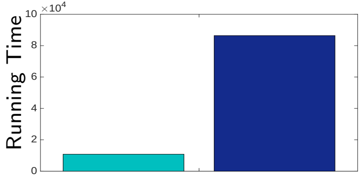


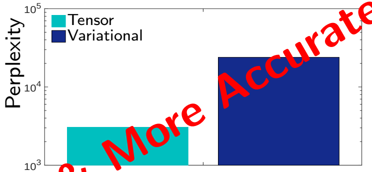
F. Huang, U.N. Niranjan, M. Hakeem, A, "Online tensor methods for training latent variable models," JMLR 2014.

Tensors vs. Variational Inference

Criterion: Perplexity $= \exp[-\text{likelihood}]$.

Learning Topics from PubMed on Spark, 8mil articles

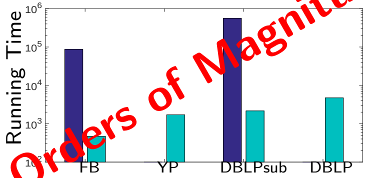


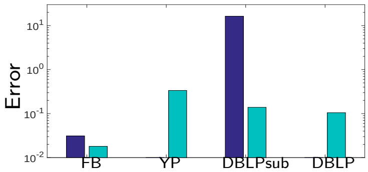


Orders of Magnitude Faster & More Accurate

Learning network communities on single workstation

Facebook $n \sim 20k$, Yelp $n \sim 40k$, DBLP-sub $n \sim 1e5$, DBLP $n \sim 1e6$.



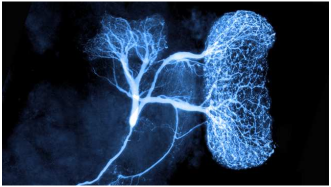


F. Huang, U.N. Niranjan, M. Hakeem, A, "Online tensor methods for training latent variable models," JMLR 2014.

Learning Representations

Sparse coding prevalent in neural signaling.

Neural sparse coding [Papadopoulou11]



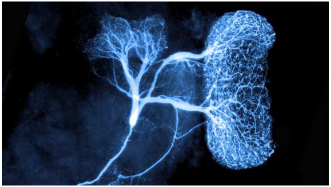
A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

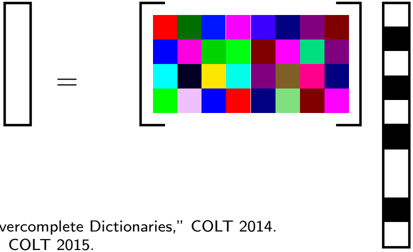
Learning Representations

Sparse coding prevalent in neural signaling.

Neural sparse coding [Papadopoulou11]



Linear Model with Overcomplete Dictionary



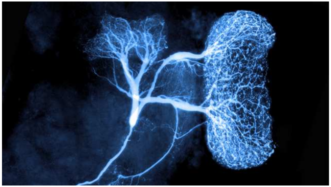
A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

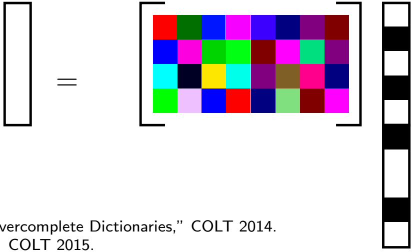
Learning Representations

Contribution: learn overcomplete incoherent dictionaries

Neural sparse coding [Papadopoulou11]



Linear Model with Overcomplete Dictionary

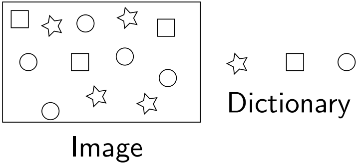


A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

Learning Representations

Shift-invariant Dictionary

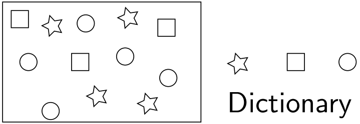


A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

Learning Representations

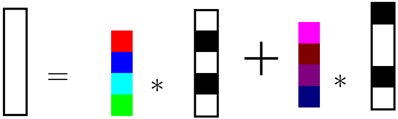
Shift-invariant Dictionary



Image

Dictionary

Convolutional Model

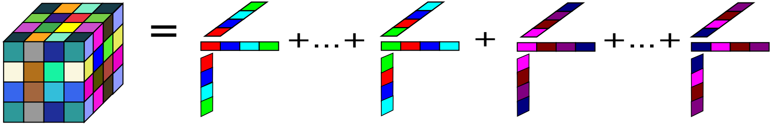


A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

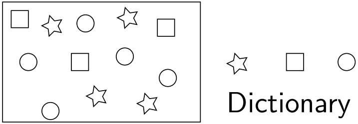
Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

Learning Representations

Efficient Tensor Decomposition with Shifted Components



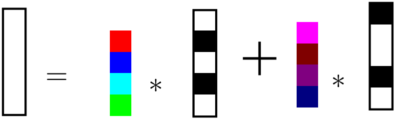
Shift-invariant Dictionary



Image

Dictionary

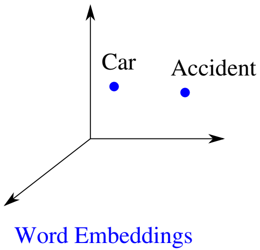
Convolutional Model



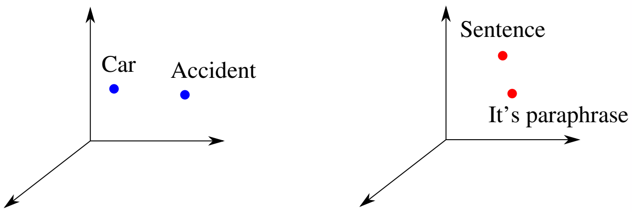
A. Agarwal, A, P. Jain, P. Netrapalli, "Learning Sparsely Used Overcomplete Dictionaries," COLT 2014.
A, M. Janzamin, R. Ge, "Overcomplete Tensor Decomposition, " COLT 2015.

Huang, A., "Convolutional Dictionary Learning through Tensor Factorization", Proc. of JMLR 2015.

Fast Text Embeddings through Tensor Methods



Fast Text Embeddings through Tensor Methods

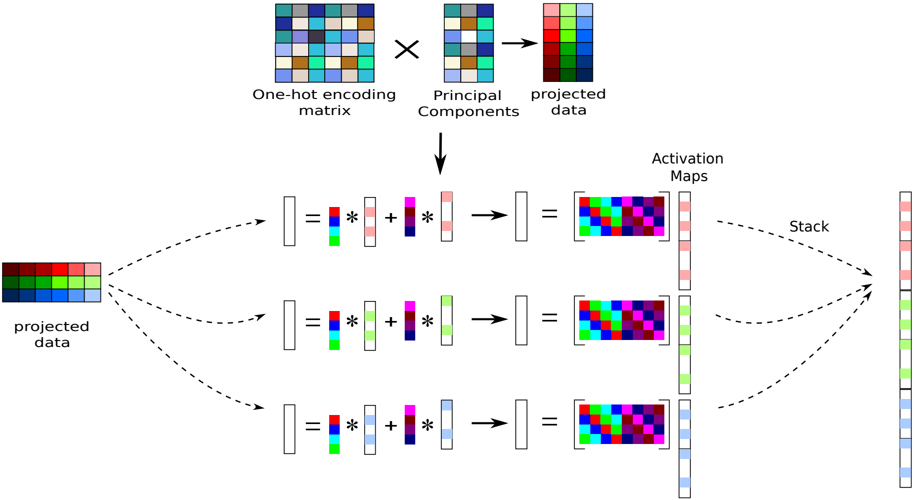


Word Embeddings

Sentence Embeddings

Fast Text Embeddings through Tensor Methods

Paraphrase Detection on MSR corpus with ~ 5000 Sentences



Fast Text Embeddings through Tensor Methods

Paraphrase Detection on MSR corpus with ~ 5000 Sentences

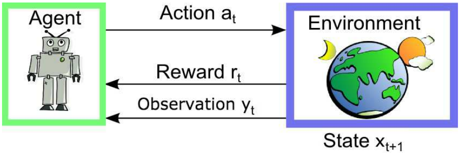
Method	F score	No. of samples
Vector Similarity (Baseline)	75%	$\sim 4k$
Tensor (Proposed)	**81%**	$\sim 4k$
Skipthought (RNN)	82%	$\sim$ 74mil

- Unsupervised learning of embeddings.
- No outside info for tensor vs. large book corpus (74 million) for skipthought
- Similar story with holographic embeddings for knowledge bases by M. Nickel et al.

Reinforcement Learning of Partially Observable Markov Decision Process

Learning in Adaptive Environments

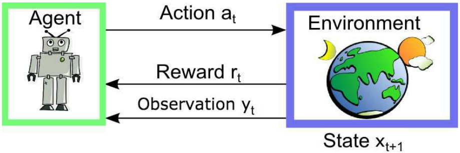
- Learner changes environment
- Hidden state estimation.



Reinforcement Learning of Partially Observable Markov Decision Process

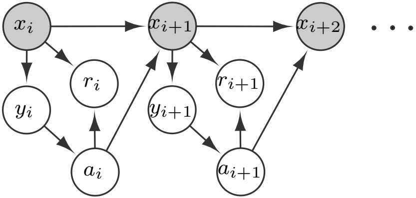
Learning in Adaptive Environments

- Learner changes environment
- Hidden state estimation.



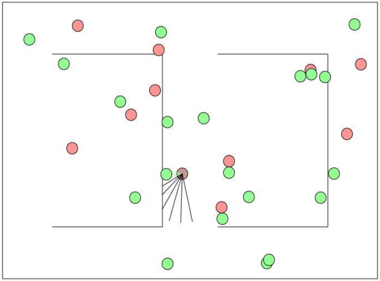
Partially Observable Markov Decision Process

- Design of tensor algorithms under memoryless policies
- Guaranteed regret bounds: comparable to fully observed environment.

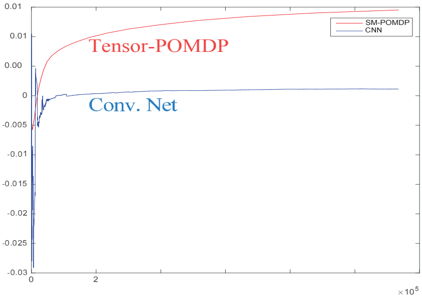


Reinforcement Learning of Partially Observable Markov Decision Process

Playing Atari Game

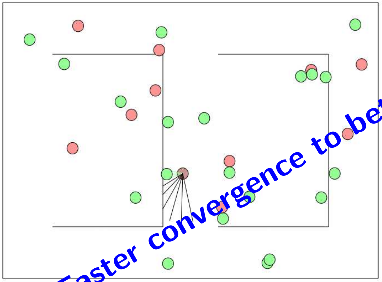


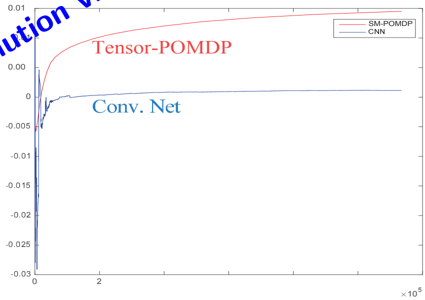
Average Reward vs. Time.



K. Azizzadenesheli, A. Lazaric, A, "Reinforcement Learning of POMDPs using Spectral Methods," 2016.

Reinforcement Learning of Partially Observable Markov Decision Process





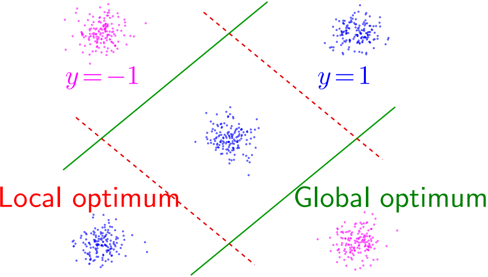
Faster convergence to better solution via tensor methods.

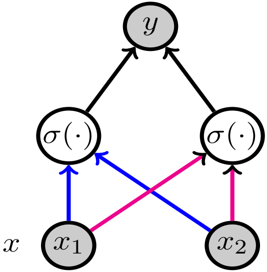
K. Azizzadenesheli, A. Lazaric, A, "Reinforcement Learning of POMDPs using Spectral Methods," 2016.

Outline

Local Optima in Backpropagation

"..few researchers dare to train their models from scratch.. small miscalibration of initial weights leads to vanishing or exploding gradients.. poor convergence..*"





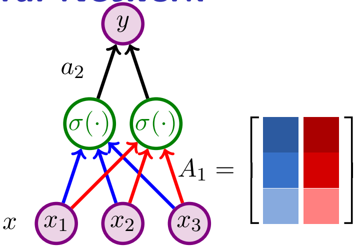
Exponential (in dimensions) no. of local optima for backpropagation

(*)

P. Krahenbhl, C. Doersch, J. Donahue, T. Darrell "Data-dependent Initializations of Convolutional Neural Networks", ICLR 2016.

Moments of a Neural Network

$$\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle$$

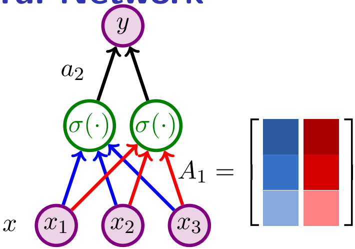


"Score Function Features for Discriminative Learning: Matrix and Tensor Framework" by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

Moments of a Neural Network

$$\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle$$

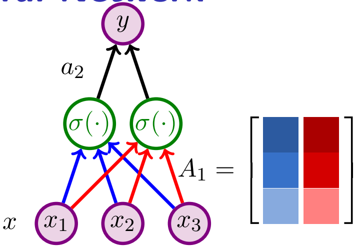
Moments using score functions $\mathcal{S}(\cdot)$



"Score Function Features for Discriminative Learning: Matrix and Tensor Framework" by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

Moments of a Neural Network

$$\boxed{\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle}$$



Moments using score functions $\mathcal{S}(\cdot)$

$$\mathbb{E}\left[y \cdot \mathcal{S}_1(x)\right] =$$

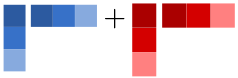
“Score Function Features for Discriminative Learning: Matrix and Tensor Framework” by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

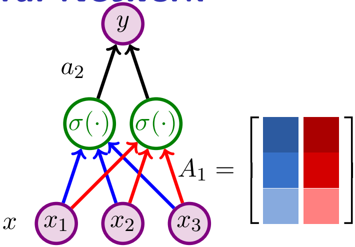
Moments of a Neural Network

$$\boxed{\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle}$$

Moments using score functions $\mathcal{S}(\cdot)$

$$\mathbb{E}\left[y \cdot \mathcal{S}_2(x)\right] =$$





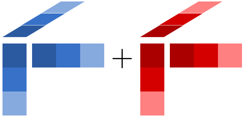
"Score Function Features for Discriminative Learning: Matrix and Tensor Framework" by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

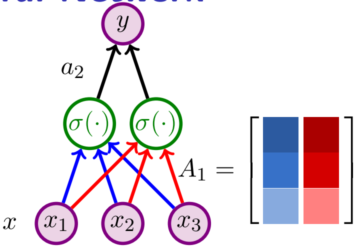
Moments of a Neural Network

$$\boxed{\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle}$$

Moments using score functions $\mathcal{S}(\cdot)$

$$\mathbb{E}\left[y \cdot \mathcal{S}_3(x)\right] =$$

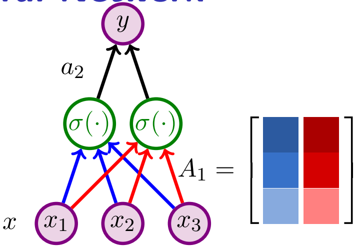




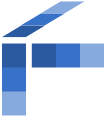
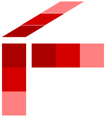
"Score Function Features for Discriminative Learning: Matrix and Tensor Framework" by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

Moments of a Neural Network

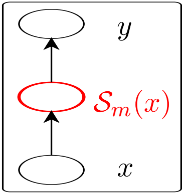
$$\boxed{\mathbb{E}[y|x] := f(x) = \langle a_2, \sigma(A_1^\top x)\rangle}$$



Moments using score functions $\mathcal{S}(\cdot)$

$\mathbb{E}\left[y \cdot \mathcal{S}_3(x)\right] =$  $+$ 

Given input pdf $p(\cdot)$, $\boxed{\mathcal{S}_m(x) := (-1)^m \dfrac{\nabla^{(m)} p(x)}{p(x)}}$.



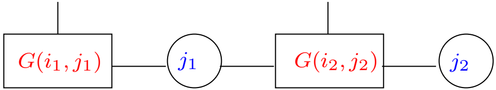
Gaussian $x \Rightarrow$ Hermite polynomials.

"Score Function Features for Discriminative Learning: Matrix and Tensor Framework" by M. Janzamin, H. Sedghi, and A. , Dec. 2014.

Tensorizing Neural Networks

- Multi-linear representation of dense layers of CNNs.
  - Tensor train format for low rank approximation of weight matrix.
- Compact representation: solves memory problem.

$$Y(i_1, i_2 \ldots) = \sum_{j_1, j_2 \ldots} G(i_1, j_1) G(i_2, j_2) \ldots X(j_1, j_2 \ldots)$$



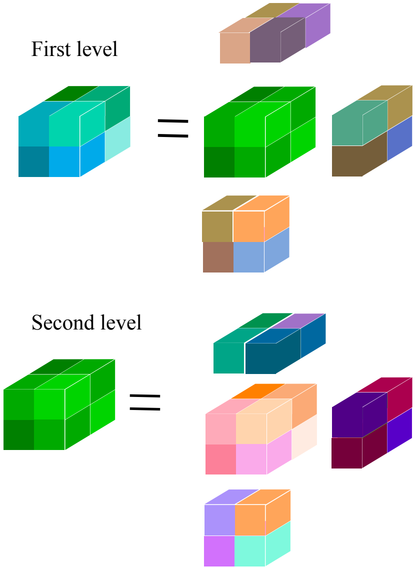
Results on ImageNet

- Compression rate $200,000$!
- Negligible performance loss.

A. Novikov, D. Podoprikhin, A. Osokin, D. Vetrov, "Tensorizing Neural Networks", NIPS 2015.

Tensor Analysis for Expressive Power

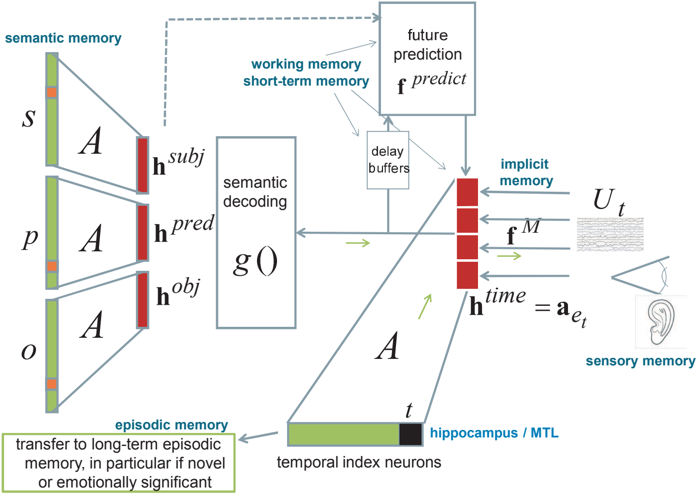
- Hierarchical Tucker tensors for representing arithmetic conv nets.
- Employs locality, sharing and pooling.
- Exponentially more parameters in shallow net vs. deep net.



N. Cohen, O. Sharir, A. Shashua, "On the Expressive Power of Deep Learning: A Tensor Analysis" COLT 2016.

Tensors in Memory Embeddings

Human Memory Model. Semantic decoding through Tensor Tucker.



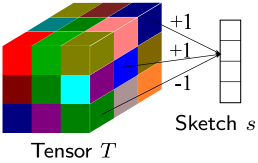
V. Tresp, C. Esteban, Y. Yang, S. Baier and D. Krompab, "Learning with Memory Embeddingss ", 2015

Outline

Scaling up and Deploying Tensor Methods

Scaling up

- Dimensionality reduction through sketching.
- Communication efficient methods.



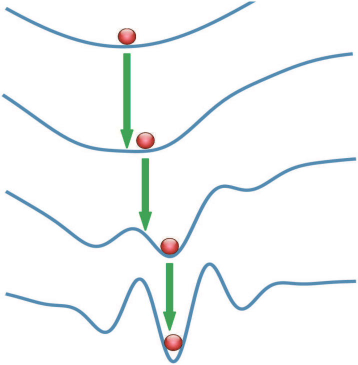
Deployment

- Multi-platform support: CPU, GPU, Cloud, FPGA ...
- Extended BLAS kernels: Beyond linear algebra.
- Many deep learning operations involve tensor contractions.

Wang, Tung, Smola, A. " Guaranteed Tensor Decomposition via Sketching", NIPS'15.

Cecka, Niranjan, Shi, A," Tensor Contractions with Extended BLAS kernels on CPU and GPU", under preparation.

Innovations in Non-Convex Methods



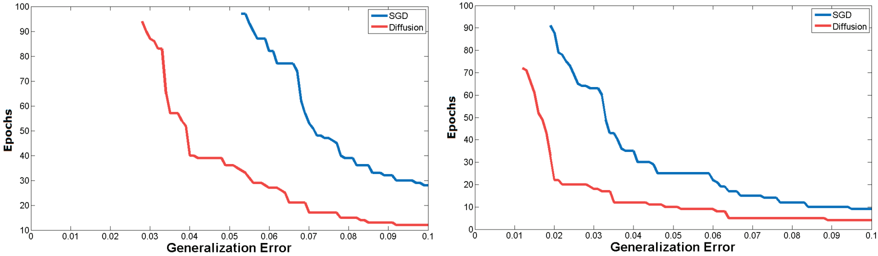
Smoothing and Continuation Methods

- Global approach vs. local search.
- Unified guarantees for non-convex problems?

H. Mobahi, "Training RNNs by Diffusion" .

Innovations in Non-Convex Methods

Learning to add using RNN

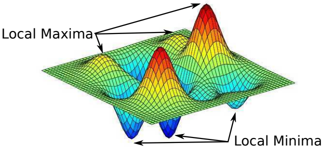


H. Mobahi, "Training RNNs by Diffusion" .

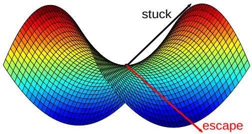
Innovations in Non-Convex Methods

- Escaping saddle points in high dimensions?
- Can SGD escape in bounded time?
- Degeneracy of saddle points in various non-convex problems?

Local optima



Saddle points



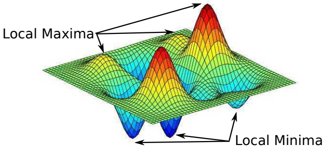
Efficient approaches for escaping higher order saddle points in non-convex optimization by A. , R. Ge, COLT 2016.

Innovations in Non-Convex Methods

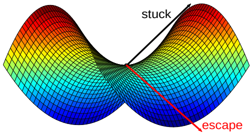
Contribution: First method to escape third order saddle

- Escaping saddle points in high dimensions?
- Can SGD escape in bounded time?
- Degeneracy of saddle points in various non-convex problems?

Local optima



Saddle points



Efficient approaches for escaping higher order saddle points in non-convex optimization by A. , R. Ge, COLT 2016.

Research Connections and Resources

Collaborators

Jennifer Chayes, Christian Borgs, Prateek Jain, Alekh Agarwal & Praneeth Netrapalli (MSR), Srinivas Turaga (Janelia), Michael Hawrylycz & Ed Lein (Allen Brain), Allesandro Lazaric (Inria), Alex Smola (CMU), Rong Ge (Duke), Daniel Hsu (Columbia), Sham Kakade (UW), Hossein Mobahi (MIT).

- Podcast/lectures/papers/software available at `http://newport.eecs.uci.edu/anandkumar/`