Implicit Regularization in Tensor Factorization

Noam Razin

Joint work with

Asaf Maman

Nadav Cohen

Tel Aviv University

Outline

1. Implicit Regularization in Deep Learning
2. Tensor Factorization
3. Implicit Tensor Rank Minimization
4. Tensor Rank as Measure of Complexity
5. Conclusion

Generalization via Bias-Variance Tradeoff

Classically, generalization is understood via the bias-variance tradeoff

Generalization via Bias-Variance Tradeoff

Classically, generalization is understood via the bias-variance tradeoff

Tradeoff can be controlled through:

Generalization via Bias-Variance Tradeoff

Classically, generalization is understood via the bias-variance tradeoff

Tradeoff can be controlled through:

- Limiting model size

Generalization via Bias-Variance Tradeoff

Classically, generalization is understood via the bias-variance tradeoff

Tradeoff can be controlled through:

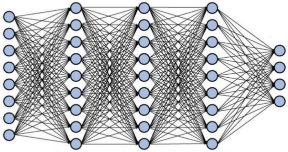
- Limiting model size
- Adding regularization (e.g. ℓ_2 penalty)

Generalization in Deep Learning

Generalization in Deep Learning

Deep neural networks (NNs) are typically overparameterized

# of learned weights



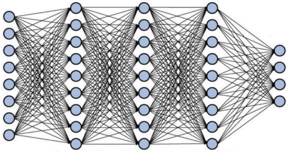
$\gg$

# of training examples

Generalization in Deep Learning

Deep neural networks (NNs) are typically overparameterized

# of learned weights



≫

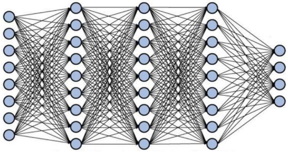
# of training examples

Can be trained with little or no regularization

Generalization in Deep Learning

Deep neural networks (NNs) are typically overparameterized

# of learned weights

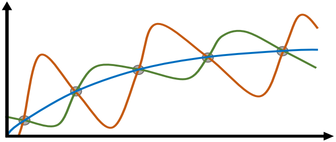


$\gg$

# of training examples

Can be trained with little or no regularization

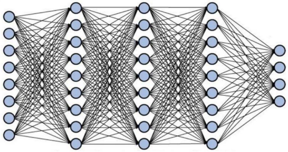
$\Longrightarrow$ Many solutions (predictors) fit training data



Generalization in Deep Learning

Deep neural networks (NNs) are typically overparameterized

# of learned weights

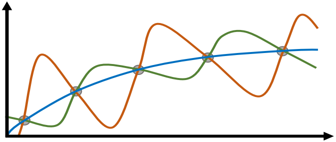


$\gg$

# of training examples

Can be trained with little or no regularization

$\Longrightarrow$ Many solutions (predictors) fit training data

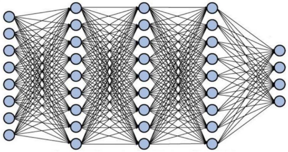


Variants of gradient descent (GD) usually find one of these solutions

Generalization in Deep Learning

Deep neural networks (NNs) are typically overparameterized

# of learned weights

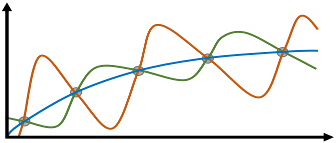


$\gg$

# of training examples

Can be trained with little or no regularization

$\Longrightarrow$ Many solutions (predictors) fit training data



Variants of gradient descent (GD) usually find one of these solutions

With "natural" data solution found often generalizes well

Conventional Wisdom: Implicit Regularization

Conventional Wisdom

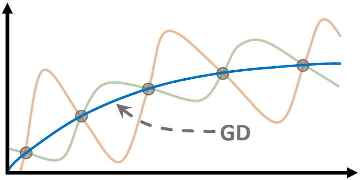
Implicit regularization minimizes "complexity":

Conventional Wisdom: Implicit Regularization

Conventional Wisdom

Implicit regularization minimizes "complexity":

- GD fits training data with predictor of lowest possible complexity

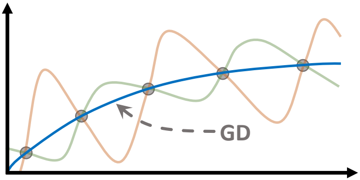


Conventional Wisdom: Implicit Regularization

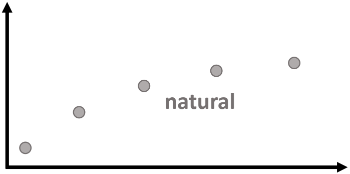
Conventional Wisdom

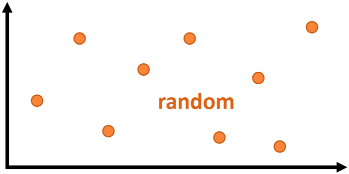
Implicit regularization minimizes "complexity":

- GD fits training data with predictor of lowest possible complexity



- Natural data can be fit with low complexity, other data cannot





Challenge: Formalizing Notion of Complexity

Goal

Mathematically formalize implicit regularization in deep learning (DL)

Challenge: Formalizing Notion of Complexity

Goal

Mathematically formalize implicit regularization in deep learning (DL)

Challenge

We lack definitions for predictor complexity that are:

Challenge: Formalizing Notion of Complexity

Goal

Mathematically formalize implicit regularization in deep learning (DL)

Challenge

We lack definitions for predictor complexity that are:

- Quantitative (admit generalization bounds)

$$\text{test error} \leq \text{train error} + \mathcal{O}\Big(\text{complexity} \,/\, \#\ \text{train examples}\Big)$$

Challenge: Formalizing Notion of Complexity

Goal

Mathematically formalize implicit regularization in deep learning (DL)

Challenge

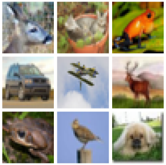
We lack definitions for predictor complexity that are:

- Quantitative (admit generalization bounds)

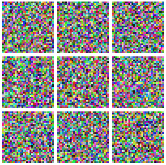
$$\text{test error} \leq \text{train error} + \mathcal{O}\Big(\text{complexity} \,/\, \#\ \text{train examples}\Big)$$

- Capture essence of natural data (allow its fit with low complexity)

✔ **low complexity**



✘ **high complexity**



Common Complexity Measures Are Insufficient

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

✔

E.g. Bartlett & Mendelson 2002, Neyshabur et al. 2015, Bartlett et al. 2017, Neyshabur et al. 2018, Golowich et al. 2018

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

  ✔

  E.g. Bartlett & Mendelson 2002, Neyshabur et al. 2015, Bartlett et al. 2017, Neyshabur et al. 2018, Golowich et al. 2018

- Captures essence of natural data (allow its fit with low complexity)?

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

  ✔

  E.g. Bartlett & Mendelson 2002, Neyshabur et al. 2015, Bartlett et al. 2017, Neyshabur et al. 2018, Golowich et al. 2018

- Captures essence of natural data (allow its fit with low complexity)?

  ✘

  E.g. Dziugaite & Roy 2017, Neyshabur et al. 2017, Jiang et al. 2020

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

✔

E.g. Bartlett & Mendelson 2002, Neyshabur et al. 2015, Bartlett et al. 2017, Neyshabur et al. 2018, Golowich et al. 2018

- Captures essence of natural data (allow its fit with low complexity)?

✘

E.g. Dziugaite & Roy 2017, Neyshabur et al. 2017, Jiang et al. 2020

When fitting data the norm is not low/margin is not high enough

Common Complexity Measures Are Insufficient

Commonly studied complexity measure: norm/margin

- Quantitative (admits generalization bounds)?

  ✔

  E.g. Bartlett & Mendelson 2002, Neyshabur et al. 2015, Bartlett et al. 2017, Neyshabur et al. 2018, Golowich et al. 2018

- Captures essence of natural data (allow its fit with low complexity)?

  ✘

  E.g. Dziugaite & Roy 2017, Neyshabur et al. 2017, Jiang et al. 2020

When fitting data the norm is not low/margin is not high enough

$\Longrightarrow$ existing generalization bounds are typically uninformative

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

				
Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

				
Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

				
Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

$d \times d'$ matrix completion $\longleftrightarrow$ prediction from $\{1,\ldots,d\} \times \{1,\ldots,d'\}$ to $\mathbb{R}$

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

				
Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

$d \times d'$ matrix completion $\longleftrightarrow$ prediction from $\{1, ..., d\} \times \{1, ..., d'\}$ to $\mathbb{R}$

value of entry (i, j) $\longleftrightarrow$ label of input (i, j)

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

$d \times d'$ matrix completion $\longleftrightarrow$ prediction from $\{1, ..., d\} \times \{1, ..., d'\}$ to $\mathbb{R}$

value of entry (i, j) $\longleftrightarrow$ label of input (i, j)

observed entries $\longleftrightarrow$ train data

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

$d \times d'$ matrix completion $\longleftrightarrow$ prediction from $\{1, ..., d\} \times \{1, ..., d'\}$ to $\mathbb{R}$

value of entry (i, j) $\longleftrightarrow$ label of input (i, j)

observed entries $\longleftrightarrow$ train data

unobserved entries $\longleftrightarrow$ test data

Matrix Completion $\longleftrightarrow$ Two-Dimensional Prediction

Matrix completion: recover unknown matrix given subset of entries

				
Bob	4	?	?	4
Alice	?	5	4	?
Joe	?	5	?	?

observations $\{y_{ij}\}_{(i,j)\in\Omega}$

Complexity measure: matrix rank

(1) Admits generalization bounds **(2)** Natural data is often low rank

$d \times d'$ matrix completion $\longleftrightarrow$ prediction from $\{1, ..., d\} \times \{1, ..., d'\}$ to $\mathbb{R}$

value of entry (i, j)	$\longleftrightarrow$	label of input (i, j)
observed entries	$\longleftrightarrow$	train data
unobserved entries	$\longleftrightarrow$	test data
matrix	$\longleftrightarrow$	predictor

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix factorization (MF):

Parameterize solution as product of matrices and fit observations via GD

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

hidden dims do not necessarily constrain rank

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix factorization (MF):

Parameterize solution as product of matrices and fit observations via GD

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

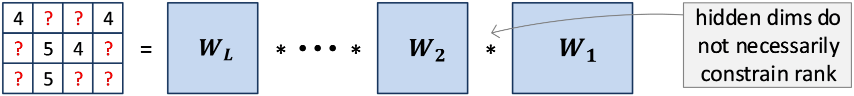
hidden dims do not necessarily constrain rank

$$\min_{W_1,\ldots,W_L} \sum\nolimits_{(i,j)\in\Omega} \ell\big([W_L W_{L-1} \cdots W_1]_{ij} - y_{ij}\big)$$

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix factorization (MF):

Parameterize solution as product of matrices and fit observations via GD



$$\min_{W_1,\ldots,W_L} \sum\nolimits_{(i,j)\in\Omega} \ell\left([W_L W_{L-1} \cdots W_1]_{ij} - y_{ij}\right)$$

$\uparrow$

Predetermined loss function (e.g. ℓ_2, ℓ_1, Huber)

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix factorization (MF):

Parameterize solution as product of matrices and fit observations via GD

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

hidden dims do not necessarily constrain rank

$$\min_{W_1,\ldots,W_L} \sum\nolimits_{(i,j)\in\Omega} \ell\big([W_L W_{L-1} \cdots W_1]_{ij} - y_{ij}\big)$$

$\uparrow$

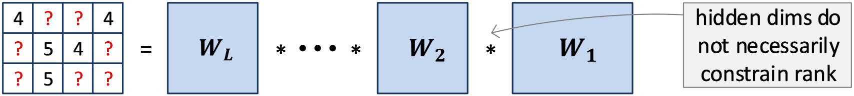
Predetermined loss function (e.g. ℓ_2, ℓ_1, Huber)

MF $\longleftrightarrow$ matrix completion via linear NN (with no explicit regularization!)

Matrix Factorization $\longleftrightarrow$ Linear Neural Network

Matrix factorization (MF):

Parameterize solution as product of matrices and fit observations via GD



$$\min_{W_1,\ldots,W_L} \sum_{(i,j)\in\Omega} \ell([W_L W_{L-1} \cdots W_1]_{ij} - y_{ij})$$

$\uparrow$

Predetermined loss function (e.g. ℓ_2, ℓ_1, Huber)

MF $\longleftrightarrow$ matrix completion via linear NN (with no explicit regularization!)

<u>**Empirical Phenomenon**</u> *(Gunasekar et al. 2017)*

MF (with small init and step size) accurately recovers low rank matrices

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Conjecture *(Gunasekar et al. 2017)*

GD over MF converges to min nuclear norm solution (predictor)

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Conjecture *(Gunasekar et al. 2017)*

GD over MF converges to min nuclear norm solution (predictor)

Dynamical Analyses

Established bias to low rank instead:

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Conjecture *(Gunasekar et al. 2017)*

GD over MF converges to min nuclear norm solution (predictor)

Dynamical Analyses

Established bias to low rank instead:

- Settings where all norms $\to \infty$ while rank is minimized *(Razin & Cohen 2020)*

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Conjecture *(Gunasekar et al. 2017)*

GD over MF converges to min nuclear norm solution (predictor)

Dynamical Analyses

Established bias to low rank instead:

- Settings where all norms $\rightarrow \infty$ while rank is minimized *(Razin & Cohen 2020)*
- Incremental rank learning *(e.g. Arora et al. 2019, Li et al. 2021)*

Implicit Regularization in Matrix Factorization

Implicit Regularization (with small init and step size):

Conjecture *(Gunasekar et al. 2017)*

GD over MF converges to min nuclear norm solution (predictor)

Dynamical Analyses

Established bias to low rank instead:

- Settings where all norms $\to \infty$ while rank is minimized *(Razin & Cohen 2020)*
- Incremental rank learning *(e.g. Arora et al. 2019, Li et al. 2021)*

Implicit regularization to low rank + data is low rank
$\Longrightarrow$ generalization

Drawbacks of Studying Matrix Factorization (MF)

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = W_L * \cdots * W_2 * W_1$$

As a surrogate for deep learning, MF is inherently limited:

Drawbacks of Studying Matrix Factorization (MF)

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

As a surrogate for deep learning, MF is inherently limited:

(1) Misses crucial aspect of non-linearity

Drawbacks of Studying Matrix Factorization (MF)

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

As a surrogate for deep learning, MF is inherently limited:

(1) Misses crucial aspect of non-linearity

(2) Does not capture prediction with more than 2 input variables

Drawbacks of Studying Matrix Factorization (MF)

$$\begin{bmatrix} 4 & ? & ? & 4 \\ ? & 5 & 4 & ? \\ ? & 5 & ? & ? \end{bmatrix} = \boldsymbol{W_L} * \cdots * \boldsymbol{W_2} * \boldsymbol{W_1}$$

As a surrogate for deep learning, MF is inherently limited:

(1) Misses crucial aspect of non-linearity

(2) Does not capture prediction with more than 2 input variables

We study tensor factorization — accounts for both **(1)** and **(2)**

Outline

1. Implicit Regularization in Deep Learning
2. Tensor Factorization
3. Implicit Tensor Rank Minimization
4. Tensor Rank as Measure of Complexity
5. Conclusion

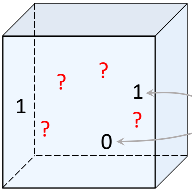
Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor completion: recover unknown tensor given subset of entries

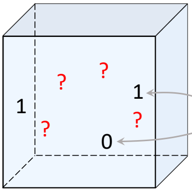


observations $\left\{y_{i_1,\ldots,i_N}\right\}_{(i_1,\ldots,i_N)\in\Omega}$

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor completion: recover unknown tensor given subset of entries



observations $\{y_{i_1,\ldots,i_N}\}_{(i_1,\ldots,i_N)\in\Omega}$

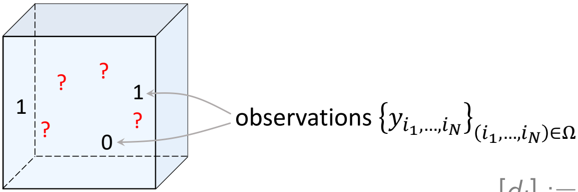
$[d_j] := \{1, \ldots, d_j\}$

$d_1 \times \cdots \times d_N$ tensor completion $\longleftrightarrow$ prediction from $[d_1] \times \cdots \times [d_N]$ to $\mathbb{R}$

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor completion: recover unknown tensor given subset of entries



$[d_j] := \{1, \ldots, d_j\}$

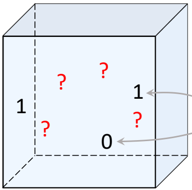
$d_1 \times \cdots \times d_N$ tensor completion $\longleftrightarrow$ prediction from $[d_1] \times \cdots \times [d_N]$ to $\mathbb{R}$

value of entry $(i_1, \ldots, i_N)$ $\longleftrightarrow$ label of input $(i_1, \ldots, i_N)$

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array (N = order of tensor)

Tensor completion: recover unknown tensor given subset of entries



observations $\{y_{i_1,\ldots,i_N}\}_{(i_1,\ldots,i_N)\in\Omega}$

$[d_j] := \{1, \ldots, d_j\}$

$d_1 \times \cdots \times d_N$ tensor completion $\longleftrightarrow$ prediction from $[d_1] \times \cdots \times [d_N]$ to $\mathbb{R}$

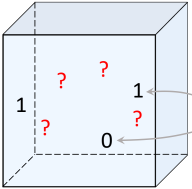
value of entry $(i_1, \ldots, i_N)$ $\longleftrightarrow$ label of input $(i_1, \ldots, i_N)$

observed entries $\longleftrightarrow$ train data

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor completion: recover unknown tensor given subset of entries



observations $\{y_{i_1,\ldots,i_N}\}_{(i_1,\ldots,i_N)\in\Omega}$

$[d_j] := \{1, \ldots, d_j\}$

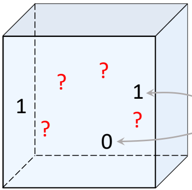
$d_1 \times \cdots \times d_N$ tensor completion $\longleftrightarrow$ prediction from $[d_1] \times \cdots \times [d_N]$ to $\mathbb{R}$

value of entry $(i_1, \ldots, i_N)$	$\longleftrightarrow$	label of input $(i_1, \ldots, i_N)$
observed entries	$\longleftrightarrow$	train data
unobserved entries	$\longleftrightarrow$	test data

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction

Tensor: N-dimensional array ($N =$ order of tensor)

Tensor completion: recover unknown tensor given subset of entries



observations $\{y_{i_1,\ldots,i_N}\}_{(i_1,\ldots,i_N)\in\Omega}$

$[d_j] := \{1,\ldots,d_j\}$

$d_1 \times \cdots \times d_N$ tensor completion $\longleftrightarrow$ prediction from $[d_1] \times \cdots \times [d_N]$ to $\mathbb{R}$

value of entry $(i_1,\ldots,i_N)$	$\longleftrightarrow$	label of input $(i_1,\ldots,i_N)$
observed entries	$\longleftrightarrow$	train data
unobserved entries	$\longleftrightarrow$	test data
tensor	$\longleftrightarrow$	predictor

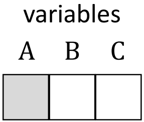
Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction (2)

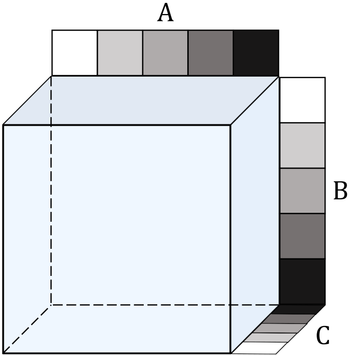
Standard prediction tasks can be seen as tensor completion problems

Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

Illustration — Image Classification (3 Pixels)

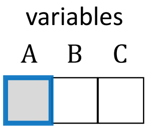


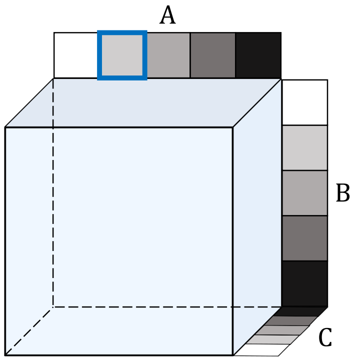


Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

Illustration — Image Classification (3 Pixels)

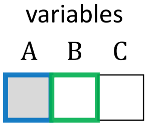


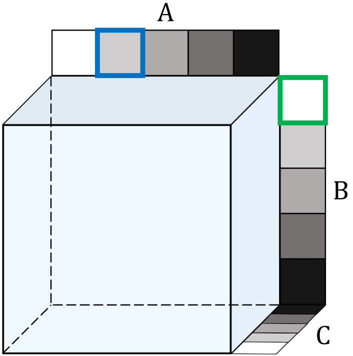


Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

Illustration — Image Classification (3 Pixels)

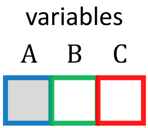


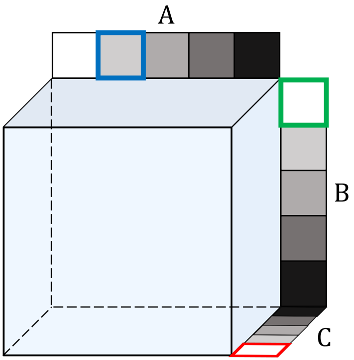


Tensor Completion $\longleftrightarrow$ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

Illustration — Image Classification (3 Pixels)

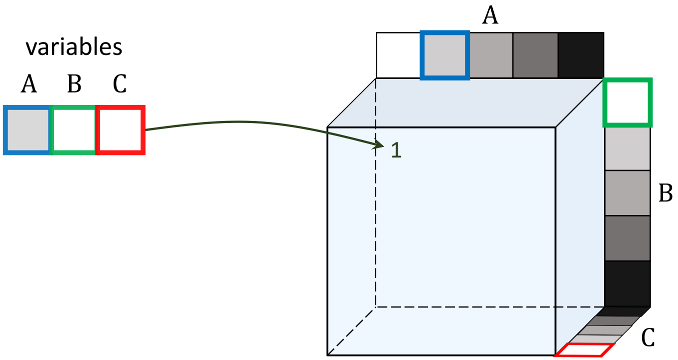




Tensor Completion ⟷ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

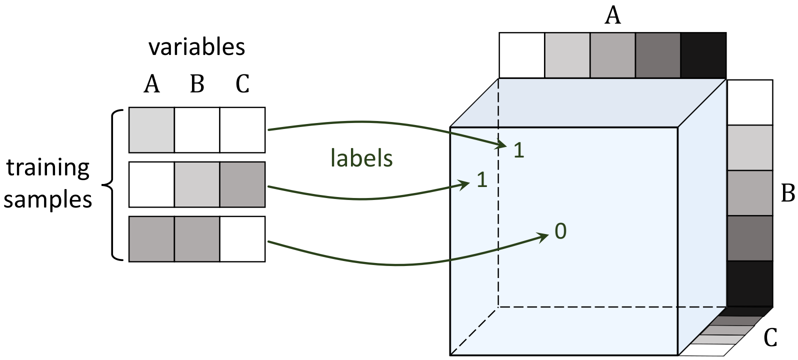
Illustration — Image Classification (3 Pixels)



Tensor Completion ⟷ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

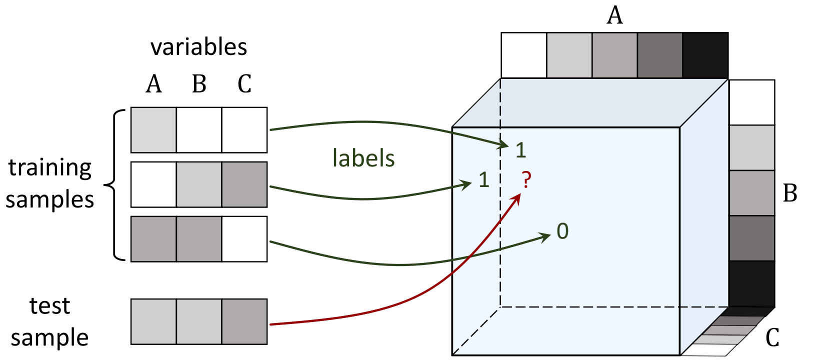
Illustration — Image Classification (3 Pixels)



Tensor Completion ⟷ Multi-Dimensional Prediction (2)

Standard prediction tasks can be seen as tensor completion problems

Illustration — Image Classification (3 Pixels)



Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor factorization (**TF**):

Parameterize solution as sum of outer products and fit observations via GD

Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor factorization (**TF**):

Parameterize solution as sum of outer products and fit observations via GD

$$\min_{\{\mathbf{w}_r^n\}_{r,n}} \sum\nolimits_{(i_1,\ldots,i_N)\in\Omega} \ell\big(\big[\sum\nolimits_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N\big]_{i_1,\ldots,i_N} - y_{i_1,\ldots,i_N}\big)$$

Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor factorization (TF):

Parameterize solution as sum of outer products and fit observations via GD

$$\min_{\{\mathbf{w}_r^n\}_{r,n}} \sum\nolimits_{(i_1,\ldots,i_N)\in\Omega} \ell\Big(\big[\sum\nolimits_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N \big]_{i_1,\ldots,i_N} - y_{i_1,\ldots,i_N} \Big)$$

Tensor rank: min # of components (R) required to express a tensor

Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor factorization (**TF**):

Parameterize solution as sum of outer products and fit observations via GD

$$\min_{\{\mathbf{w}_r^n\}_{r,n}} \sum\nolimits_{(i_1,\ldots,i_N)\in\Omega} \ell\Big(\big[\underset{\substack{\uparrow}}{\textstyle\sum_{r=1}^{R}} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N\big]_{i_1,\ldots,i_N} - y_{i_1,\ldots,i_N}\Big)$$

R large enough to not constrain rank

Tensor rank: min # of components (R) required to express a tensor

Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

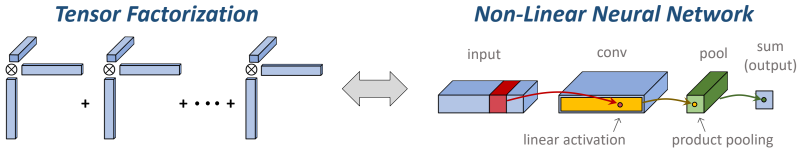
Tensor factorization (TF):

Parameterize solution as sum of outer products and fit observations via GD

$$\min_{\{\mathbf{w}_r^n\}_{r,n}} \sum\nolimits_{(i_1,\ldots,i_N)\in\Omega} \ell\Big(\big[\underset{\substack{\uparrow \\ R \text{ large enough to not constrain rank}}}{\textstyle\sum_{r=1}^{R}} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N\big]_{i_1,\ldots,i_N} - y_{i_1,\ldots,i_N}\Big)$$

Tensor rank: min # of components (R) required to express a tensor

TF $\longleftrightarrow$ tensor completion via NN with multiplicative non-linearity



Tensor Factorization $\longleftrightarrow$ Non-Linear Neural Network

Tensor factorization (TF):

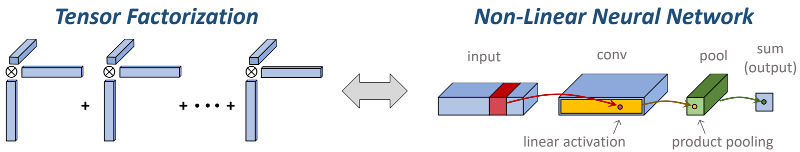
Parameterize solution as sum of outer products and fit observations via GD

$$\min_{\{\mathbf{w}_r^n\}_{r,n}} \sum\nolimits_{(i_1,\ldots,i_N)\in\Omega} \ell\Big(\big[\underbrace{\textstyle\sum_{r=1}^{R}}_{\uparrow}\mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N\big]_{i_1,\ldots,i_N} - y_{i_1,\ldots,i_N}\Big)$$

R large enough to not constrain rank

Tensor rank: min # of components (R) required to express a tensor

TF $\longleftrightarrow$ tensor completion via NN with multiplicative non-linearity



Equivalence extensively studied (e.g. Cohen et al. 2016, Levine et al. 2018, Khrulkov et al. 2018)

Implicit Regularization in Tensor Factorization

Question

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$

"end tensor"

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$

"end tensor"

Empirical Phenomenon *(Razin & Cohen 2020)*

TF (with small init and step size) accurately recovers low rank tensors

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$

"end tensor"

Empirical Phenomenon *(Razin & Cohen 2020)*

TF (with small init and step size) accurately recovers low rank tensors

Current Talk

Theoretically support empirical phenomenon

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$

"end tensor"

Empirical Phenomenon *(Razin & Cohen 2020)*

TF (with small init and step size) accurately recovers low rank tensors

Current Talk

Theoretically support empirical phenomenon

Dynamical analysis reveals that with small init and step size:

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$
"end tensor"

Empirical Phenomenon *(Razin & Cohen 2020)*

TF (with small init and step size) accurately recovers low rank tensors

Current Talk

Theoretically support empirical phenomenon

Dynamical analysis reveals that with small init and step size:

- Incremental tensor rank learning $\Longrightarrow$ bias towards low tensor rank

Implicit Regularization in Tensor Factorization

Question

To which solutions does $\mathcal{W}_e := \sum_{r=1}^{R} \mathbf{w}_r^1 \otimes \cdots \otimes \mathbf{w}_r^N$ converge?

$\uparrow$

"end tensor"

Empirical Phenomenon *(Razin & Cohen 2020)*

TF (with small init and step size) accurately recovers low rank tensors

Current Talk

Theoretically support empirical phenomenon

Dynamical analysis reveals that with small init and step size:

- Incremental tensor rank learning $\Longrightarrow$ bias towards low tensor rank
- Tensor rank minimization (under technical conditions)

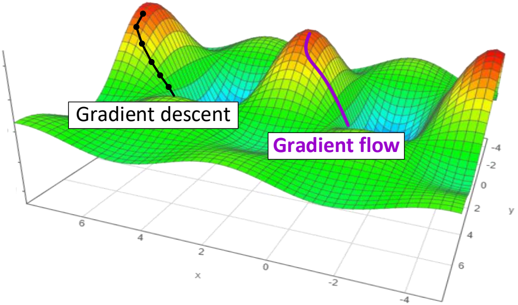
Outline

1. Implicit Regularization in Deep Learning
2. Tensor Factorization
3. Implicit Tensor Rank Minimization
4. Tensor Rank as Measure of Complexity
5. Conclusion

Gradient Flow

Gradient flow (GF) is a continuous version of GD (step size $\rightarrow 0$):

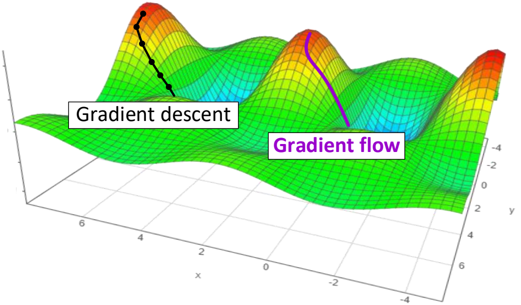
$$\frac{d}{dt}\theta(t) = -\nabla f(\theta(t)) \quad , \; t \geq 0$$



Gradient Flow

Gradient flow (GF) is a continuous version of GD (step size $\to 0$):

$$\frac{d}{dt}\theta(t) = -\nabla f(\theta(t)) \quad , \; t \geq 0$$

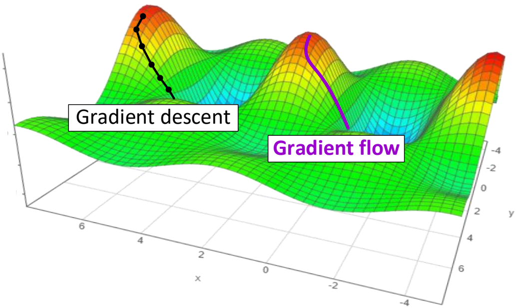


Admits use of theoretical tools from differential geometry/equations

Gradient Flow

Gradient flow (GF) is a continuous version of GD (step size $\to 0$):

$$\frac{d}{dt}\theta(t) = -\nabla f(\theta(t)) \quad , \; t \geq 0$$



Admits use of theoretical tools from differential geometry/equations

Closely matches GD in practice for tensor factorization

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2 - \frac{2}{N} > 1$

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2-\frac{2}{N} > 1$
- Components move slower when small and faster when large

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2-\frac{2}{N} > 1$
- Components move slower when small and faster when large
- Small init

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2 - \frac{2}{N} > 1$
- Components move slower when small and faster when large
- Small init $\implies$ Incremental growth of components

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2 - \frac{2}{N} > 1$
- Components move slower when small and faster when large
- Small init $\implies$ Incremental growth of components $\implies$ low tensor rank

Dynamical Analysis of Implicit Regularization

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Interpretation

- Order $N \geq 3 \implies$ the exponent $2 - \frac{2}{N} > 1$
- Components move slower when small and faster when large
- Small init $\implies$ Incremental growth of components $\implies$ low tensor rank

Generalizes existing characterization for matrix factorization

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N}\mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N}\mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N}\mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor , $\mathcal{L}(\cdot) :=$ loss w.r.t. $\mathcal{W}_e$

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N}\mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor , $\mathcal{L}(\cdot) :=$ loss w.r.t. $\mathcal{W}_e$, $\widehat{\mathbf{w}}_r^n := \frac{\mathbf{w}_r^n}{\|\mathbf{w}_r^n\|}$

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\|\mathbf{w}_r^n(t)\|^2 - \|\mathbf{w}_r^{\bar{n}}(t)\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N}\mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor , $\mathcal{L}(\cdot) :=$ loss w.r.t. $\mathcal{W}_e$, $\widehat{\mathbf{w}}_r^n := \frac{\mathbf{w}_r^n}{\|\mathbf{w}_r^n\|}$

Differentiating w.r.t. time:

$$\frac{d}{dt}\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\| = \ldots = \sum_{n=1}^{N}\prod_{n'\neq n}\|\mathbf{w}_r^{n'}(t)\|^2 \cdot \left\langle -\nabla\mathcal{L}(\mathcal{W}_e(t)), \otimes_{n=1}^{N}\widehat{\mathbf{w}}_r^n(t)\right\rangle$$

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N} \mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor , $\mathcal{L}(\cdot) :=$ loss w.r.t. $\mathcal{W}_e$, $\widehat{\mathbf{w}}_r^n := \frac{\mathbf{w}_r^n}{\|\mathbf{w}_r^n\|}$

Differentiating w.r.t. time:

$$\frac{d}{dt}\|\otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| = \ldots = \sum_{n=1}^{N}\prod_{n'\neq n} \|\mathbf{w}_r^{n'}(t)\|^2 \cdot \left\langle -\nabla\mathcal{L}(\mathcal{W}_e(t)), \otimes_{n=1}^{N}\widehat{\mathbf{w}}_r^n(t)\right\rangle$$

Component Norm Dynamics Theorem — Proof Sketch

Theorem

When initialized near-zero, the norm of the r'th component evolves by:

$$\frac{d}{dt}\left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\| \propto \left\|\otimes_{n=1}^{N}\mathbf{w}_r^n(t)\right\|^{2-\frac{2}{N}}$$

Proof Sketch

For any $n, \bar{n}$: $\left|\left\|\mathbf{w}_r^n(t)\right\|^2 - \left\|\mathbf{w}_r^{\bar{n}}(t)\right\|^2\right|$ is constant through time

$\implies$ when init is small $\|\mathbf{w}_r^n(t)\|^2 \approx \|\mathbf{w}_r^{\bar{n}}(t)\|^2 \approx \|\otimes_{n'=1}^{N} \mathbf{w}_r^{n'}(t)\|^{\frac{2}{N}}$

Denote:

$\mathcal{W}_e := \sum_{r=1}^{R}\otimes_{n=1}^{N}\mathbf{w}_r^n$ — end tensor , $\mathcal{L}(\cdot) :=$ loss w.r.t. $\mathcal{W}_e$, $\widehat{\mathbf{w}}_r^n := \frac{\mathbf{w}_r^n}{\|\mathbf{w}_r^n\|}$

Differentiating w.r.t. time:

$$\frac{d}{dt}\|\otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| \approx \|\otimes_{n=1}^{N} \mathbf{w}_r^n(t)\|^{2-\frac{2}{N}} \cdot N\langle -\nabla\mathcal{L}(\mathcal{W}_e(t)), \otimes_{n=1}^{N}\widehat{\mathbf{w}}_r^n(t)\rangle$$

Dynamical Analysis — Experiments

Dynamical Analysis — Experiments

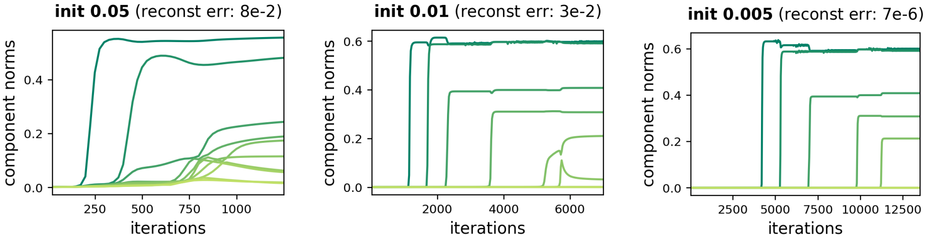
Rank 5 Order 4 Tensor Completion

Fit observations via GD over 1000-component tensor factorization

Dynamical Analysis — Experiments

Rank 5 Order 4 Tensor Completion

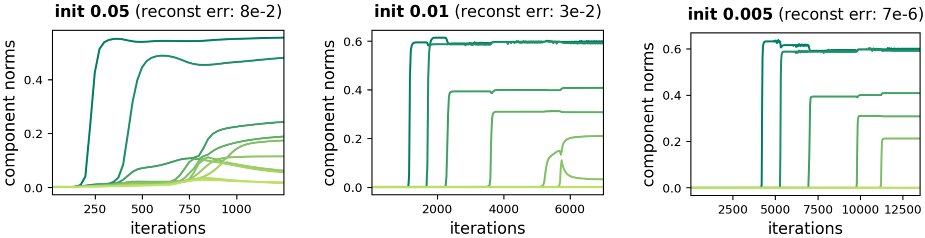
Fit observations via GD over 1000-component tensor factorization



Dynamical Analysis — Experiments

Rank 5 Order 4 Tensor Completion

Fit observations via GD over 1000-component tensor factorization

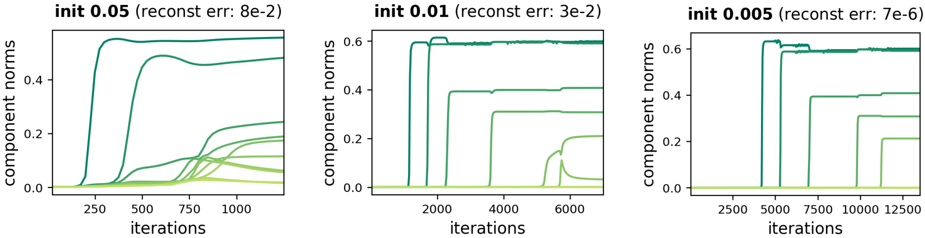


As init $\rightarrow 0$ fewer components depart from zero

Dynamical Analysis — Experiments

Rank 5 Order 4 Tensor Completion

Fit observations via GD over 1000-component tensor factorization



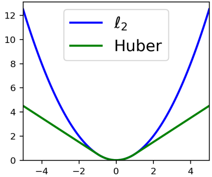
As init $\rightarrow 0$ fewer components depart from zero

Incremental learning of components leads to low tensor rank!

Implicit Tensor Rank Minimization: Rank One Trajectory

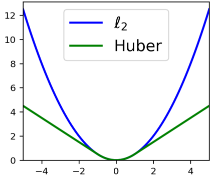
Implicit Tensor Rank Minimization: Rank One Trajectory

Assume Huber loss with no observation exactly 0



Implicit Tensor Rank Minimization: Rank One Trajectory

Assume Huber loss with no observation exactly 0

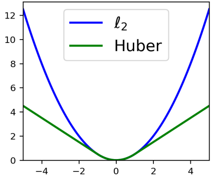


and that there exists component at init with:

(1) positive projection onto $-\nabla\mathcal{L}(0)$ **and** **(2)** norm $>$ others

Implicit Tensor Rank Minimization: Rank One Trajectory

Assume Huber loss with no observation exactly 0



and that there exists component at init with:

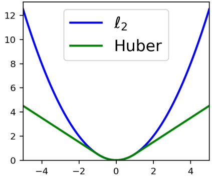
(1) positive projection onto $-\nabla\mathcal{L}(0)$ **and** **(2)** norm $>$ others

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Implicit Tensor Rank Minimization: Rank One Trajectory

Assume Huber loss with no observation exactly 0



and that there exists component at init with:

(1) positive projection onto $-\nabla \mathcal{L}(0)$ **and** **(2)** norm $>$ others

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Corollary

If rank 1 trajectories converge to $\mathcal{W}^*$, then $\mathcal{W}_e(t) \to \mathcal{W}^*$ as init $\to 0$

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt} \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t) \| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t) \|^{2 - \frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt}\| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\|^{2-\frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

$\implies$ exists time at which one component is $\Omega(1)$ while others are $\mathcal{O}(\alpha^N)$

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt}\| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\|^{2-\frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

$\implies$ exists time at which one component is $\Omega(1)$ while others are $\mathcal{O}(\alpha^N)$

Taking $\alpha \to 0$

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt}\| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\|^{2-\frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

$\implies$ exists time at which one component is $\Omega(1)$ while others are $\mathcal{O}(\alpha^N)$

Taking $\alpha \to 0$

$\implies$ at that time $\mathcal{W}_e(t)$ is arbitrarily close to a rank 1 trajectory

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt} \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t) \| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t) \|^{2-\frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

$\implies$ exists time at which one component is $\Omega(1)$ while others are $\mathcal{O}(\alpha^N)$

Taking $\alpha \to 0$

$\implies$ at that time $\mathcal{W}_e(t)$ is arbitrarily close to a rank 1 trajectory

Loss is locally smooth

Rank One Trajectory Theorem — Proof Sketch

Theorem

For any time T, distance D, and ϵ, if init is sufficiently small, $\mathcal{W}_e(t)$ is ϵ close to a rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Proof Sketch

Denote init scale $\alpha > 0$

$\frac{d}{dt}\| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\| \propto \| \otimes_{n=1}^{N} \mathbf{w}_r^n(t)\|^{2-\frac{2}{N}}$ and $\nabla \mathcal{L}(\mathcal{W}_e)$ is const around origin

$\implies$ exists time at which one component is $\Omega(1)$ while others are $\mathcal{O}(\alpha^N)$

Taking $\alpha \to 0$

$\implies$ at that time $\mathcal{W}_e(t)$ is arbitrarily close to a rank 1 trajectory

Loss is locally smooth

$\implies$ $\mathcal{W}_e(t)$ is ϵ close to rank 1 trajectory until $t \geq T$ or $\|\mathcal{W}_e(t)\| \geq D$

Outline

1. Implicit Regularization in Deep Learning
2. Tensor Factorization
3. Implicit Tensor Rank Minimization
4. Tensor Rank as Measure of Complexity
5. Conclusion

Challenge: Formalizing Notion of Complexity

Goal

Mathematically formalize implicit regularization in deep learning (DL)

Challenge

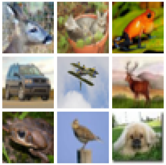
We lack definitions for predictor complexity that are:

- Quantitative (admit generalization bounds)

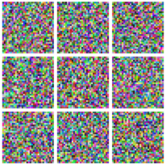
$$\text{test error} \leq \text{train error} + \mathcal{O}\Big(\text{complexity} \,/\, \#\ \text{train examples}\Big)$$

- Capture essence of natural data (allow its fit with low complexity)

✔ **low complexity**



✘ **high complexity**



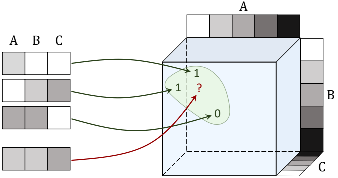
Tensor Rank Captures Non-Linear Neural Network

We saw:

Tensor Rank Captures Non-Linear Neural Network

We saw:

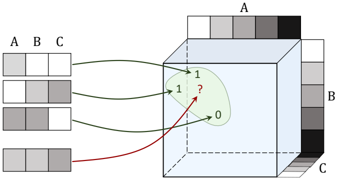
- Tensor completion $\longleftrightarrow$ multi-dimensional prediction



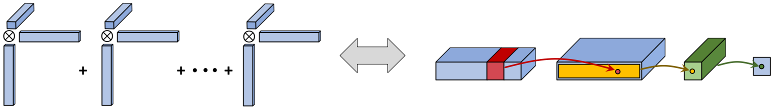
Tensor Rank Captures Non-Linear Neural Network

We saw:

- Tensor completion $\longleftrightarrow$ multi-dimensional prediction



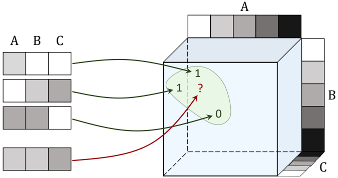
- Tensor factorization $\longleftrightarrow$ non-linear NN



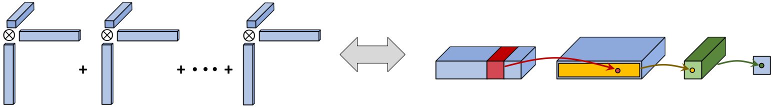
Tensor Rank Captures Non-Linear Neural Network

We saw:

- Tensor completion $\longleftrightarrow$ multi-dimensional prediction



- Tensor factorization $\longleftrightarrow$ non-linear NN

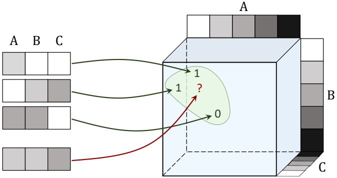


- Implicit regularization favors tensors (predictors) of low tensor rank

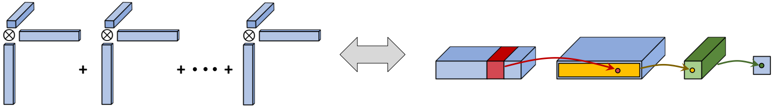
Tensor Rank Captures Non-Linear Neural Network

We saw:

- Tensor completion $\longleftrightarrow$ multi-dimensional prediction



- Tensor factorization $\longleftrightarrow$ non-linear NN



- Implicit regularization favors tensors (predictors) of low tensor rank

Question

Can tensor rank serve as measure of complexity for predictors?

Experiment: Fitting Data with Low Tensor Rank

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Datasets:

- MNIST    and Fashion-MNIST    (one-vs-all)

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Datasets:

- MNIST    and Fashion-MNIST    (one-vs-all)
- Each compared against:

  (i) random images (same labels) (ii) random labels (same images)

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Datasets:

- MNIST    and Fashion-MNIST    (one-vs-all)
- Each compared against:

  (i) random images (same labels) (ii) random labels (same images)

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Datasets:

- MNIST    and Fashion-MNIST    (one-vs-all)
- Each compared against:

  (i) random images (same labels) (ii) random labels (same images)

Original data fit far more accurately than random (leading to low test err)!

Experiment: Fitting Data with Low Tensor Rank

Experiment

Fitting standard datasets with predictors of low tensor rank

Datasets:

- MNIST  and Fashion-MNIST  (one-vs-all)
- Each compared against:

  (i) random images (same labels) (ii) random labels (same images)

Original data fit far more accurately than random (leading to low test err)!

Tensor rank may shed light on both implicit regularization of NNs and properties of real-world data translating it to generalization

Outline

1. Implicit Regularization in Deep Learning
2. Tensor Factorization
3. Implicit Tensor Rank Minimization
4. Tensor Rank as Measure of Complexity
5. Conclusion

Recap

Recap

Goal: Understanding implicit regularization in DL

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Tensor Rank as Measure of Complexity

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Tensor Rank as Measure of Complexity

Standard datasets are fitted by predictors of low tensor rank

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Tensor Rank as Measure of Complexity

Standard datasets are fitted by predictors of low tensor rank

Hypothesis

Tensor rank may pave way to understanding:

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Tensor Rank as Measure of Complexity

Standard datasets are fitted by predictors of low tensor rank

Hypothesis

Tensor rank may pave way to understanding:

- Implicit regularization of neural networks

Recap

Goal: Understanding implicit regularization in DL

- Challenge: lack measures of complexity that capture natural data

Tensor Factorization

- Equivalent to multi-dim prediction via non-linear NN
- Dynamical analysis: implicit regularization minimizes tensor rank

Tensor Rank as Measure of Complexity

Standard datasets are fitted by predictors of low tensor rank

Hypothesis

Tensor rank may pave way to understanding:

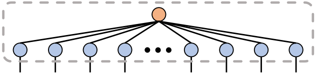
- Implicit regularization of neural networks
- Properties of data translating it to generalization

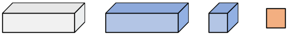
Ongoing Work: Adding Depth via Hierarchy

Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization

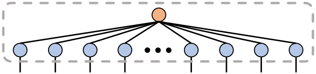
Shallow Non-Linear Neural Network

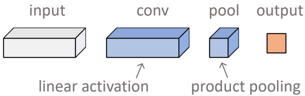




Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization ⟺ ***Shallow Non-Linear Neural Network***

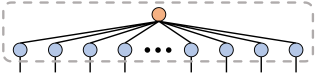




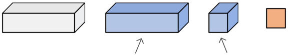
Implicit regularization = minimization of tensor rank

Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization



Shallow Non-Linear Neural Network

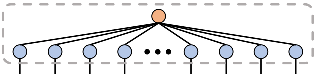


Implicit regularization = minimization of tensor rank

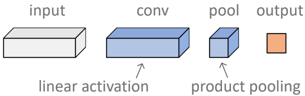
❌ **Oblivious to input ordering**

Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization



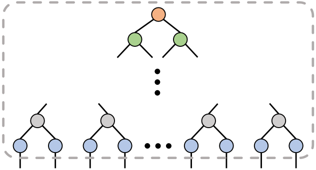
Shallow Non-Linear Neural Network



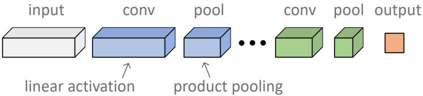
Implicit regularization = minimization of tensor rank

❌ **Oblivious to input ordering**

Hierarchical Tensor Factorization

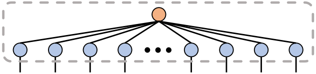


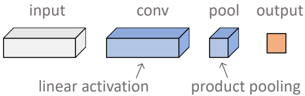
Deep Non-Linear Neural Network



Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization | ***Shallow Non-Linear Neural Network***

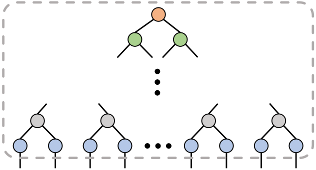


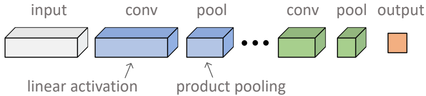


Implicit regularization = minimization of $\underbrace{\text{tensor rank}}$

❌ **Oblivious to input ordering**

Hierarchical Tensor Factorization | ***Deep Non-Linear Neural Network***

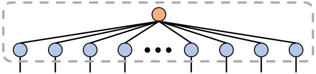


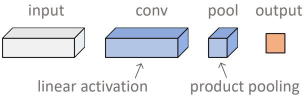


Implicit regularization $\overset{?}{=}$ minimization of hierarchical tensor rank

Ongoing Work: Adding Depth via Hierarchy

Tensor Factorization ⟷ ***Shallow Non-Linear Neural Network***

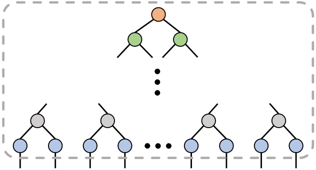


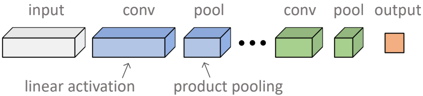


Implicit regularization = minimization of tensor rank

❌ **Oblivious to input ordering**

Hierarchical Tensor Factorization ⟷ ***Deep Non-Linear Neural Network***





Implicit regularization $\overset{?}{=}$ minimization of hierarchical tensor rank

✔ **Accounts for input ordering**

Thank You

Work supported by: Amnon and Anat Shashua, Len Blavatnik and the Blavatnik Family Foundation, Yandex Initiative in Machine Learning, Google Research Gift