UNIVERSITY OF OXFORD

Mathematical Institute

Data classes for which DNNs can overcome the curse of dimensionality.

THEORIES OF DEEP LEARNING: C6.5,
LECUTRE / VIDEO 4

Prof. Jared Tanner
Mathematical Institute
University of Oxford

Oxford Mathematics

Curse of dimensionality (Yarotsky 16')

Exponential in d/m growth in number of weights.

Yarotsky 16' results show exponential approximation in depth, but the overall number of weights is $\mathcal{O}(\epsilon^{-d/m})$. Recall

$$\|f\|_{W_m^\infty}([0,1]^d) = \max_{|s|\le m} \text{esssupp}_{x\in[0,1]^d} |D^s f(x)|.$$

Theorem (Yarotsky 16')

For any d, m and $\epsilon \in (0,1)$, there is a ReLU network with depth at most $c(1+\ln(1/\epsilon))$ and at most $c\epsilon^{-d/m}(1+\log(1/\epsilon))$ weights (width $\mathcal{O}(\epsilon^{-d/m})$), for c a function of d, m, that can approximate any function from $F_{d,m}$ within absolute error ϵ.

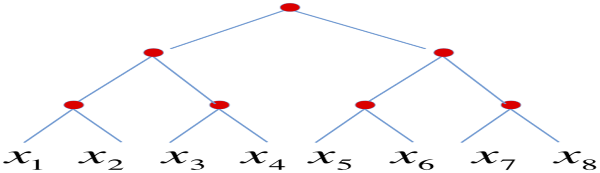
https://arxiv.org/pdf/1610.01145.pdf

To avoid curse of dimensionality need $m \sim d$ or more structure in the function F to be approximated; e.g. compositional structure.

Compositional structured functions (Poggio et al. 17')

Extending the compositional nature of Yarotsky dimensionally

Consider functions with a binary tree hierarchical structure:



where $x \in \mathbb{R}^8$ and
$f(x) = h_3(h_{21}(h_{11}(x_1, x_2), h_{12}(x_3, x_4)), h_{22}(h_{13}(x_5, x_6), h_{14}(x_7, x_8)))$
Let $W_m^{n,2}$ be the class of all compositional functions $f(\cdot)$ of n variables with binary tree structure and constituent functions $h(\cdot)$ of 2 variables with m bounded derivatives.

https://arxiv.org/pdf/1611.00740.pdf

Compositional structured functions (Poggio et al. 17')

Each constituent function is a map from $\mathbb{R}^2 \to \mathbb{R}$

The set $W_m^{n,2}$ of of all compositional functions $f(\cdot)$ of n variables with binary tree structure and constituent functions $h(\cdot)$ of 2 variables with m bounded derivatives can be effectively approximated using a DNN with a rate dictated by the ability to approximate functions $\mathbb{R}^2 \to \mathbb{R}$; e.g. effectively locally $d = 2$.

Theorem (Poggio 17')

Let $f(\cdot) \in W_m^{n,2}$ and consider a DNN with the same binary compositional tree structure and an activation $\sigma(\cdot)$ which is infinitely differentiable, and not a polynomial. The function $f(\cdot)$, can be approximated by ϵ with a number of weights that is $\mathcal{O}\left((n-1)\epsilon^{-2/m}\right)$.

`https://arxiv.org/pdf/1611.00740.pdf`

Compositional structured functions (Poggio et al. 17')

Compositional functions $W_m^{n,2}$ compared to shallow NNs and Yarotsky(16')

Mathematical Institute

The set $W_m^{n,2}$ of of all compositional functions $f(\cdot)$ of n variables with binary tree structure are effectively $d = 2$ in the DNN approximation requirements, but are much richer than $d = 2$.

Functions can be approximated within ϵ with a DNN from $\mathcal{O}(ln(1/\epsilon))$ layers with a number of weights:

- $\mathcal{O}(\epsilon^{-d/m})$ for general locally smooth functions (Yarotsky 16'),
- $\mathcal{O}\left((n-1)\epsilon^{-2/m}\right)$ for $f(\cdot) \in W_m^{n,2}$, binary tree structure and constituent functions in $C_m[0,1]^2$.
- $\mathcal{O}(\epsilon^{-d/m})$ for shallow NNs is best possible for $f(\cdot) \in W_m^n$ which have non-binary hierarchical tree structures.

`https://arxiv.org/pdf/1611.00740.pdf`

Definition of local effective dimensionality (Poggio et al. 17')

Local dimensionality determined by approximation rate ϵ^{-d}.

Definition (Poggio 17')

The effective dimensionality of a function class W is said to be d if for every $\epsilon > 0$, any function within W can be approximated within an accuracy ϵ by a DNN at rate ϵ^{-d}.

In the prior slide we had examples of complex compositional functions with effective dimensionality 2. These could be extended naturally to local *effective dimensionality* d_{eff} and *local smoothness* m_{eff} for rate $\epsilon^{-d_{eff}/m_{eff}}$.

Restriction to a data class decreases d_{eff} and localisation can increase the smoothness m_{eff} substantially.
https://arxiv.org/pdf/1611.00740.pdf

Intrinsic dimensionality of sub-manifolds (Hein et al. 05')

MNIST exemplar dataset classes are approximately under 15 dimensional

Estimates of dimensionality within MNIST digit classes using three approaches: the reference below, and two others building on local linear embedding.

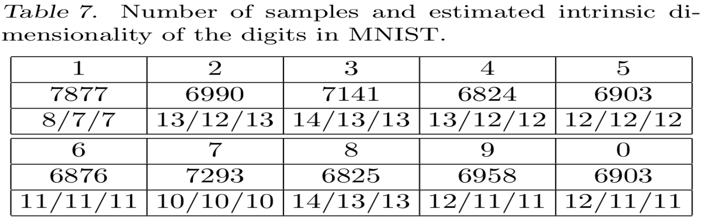


Table 7. Number of samples and estimated intrinsic dimensionality of the digits in MNIST.

1	2	3	4	5	6	7	8	9	0
7877	6990	7141	6824	6903	6876	7293	6825	6958	6903
8/7/7	13/12/13	14/13/13	13/12/12	12/12/12	11/11/11	10/10/10	14/13/13	12/11/11	12/11/11

https://icml.cc/Conferences/2005/proceedings/papers/037_Intrinsic_HeinAudibert.pdf

The hidden manifold model (Goldt et al. 19')

An alternative manifold model: one layer GAN

A manifold model can explicitly represent the data through:

$$X = f(CF/\sqrt{d}) \in \mathbb{R}^{p,n}$$

where:

- $F \in \mathbb{R}^{d,n}$ are the d features used to represent the data
- $C \in \mathbb{R}^{p,d}$ combines the $d < n < p$ features
- $f(\cdot)$ is an entrywise locally smooth nonlinear function.

This data model is the same as a generative adversarial network (GAN) and is similar to dictionary learning and subspace clustering models where C is typically sparse.

https://hal-cea.archives-ouvertes.fr/cea-02529246/document

Additional approximation theory resources

Review articles and courses elsewhere

Further references for the approximation theory perspective of deep learning include:

- Telgarsky's "Deep Learning Theory" course, lectures 1-11: `http://mjt.cs.illinois.edu/courses/dlt-f20/`
- Matthew Hirn's "Mathematics of Deep Learning" course: lectures 20-24. `https://matthewhirn.com/teaching/spring-2020-cmse-890-002/`
- DNN Approximation Theory by Elbrachter et al. (19') `https://www.mins.ee.ethz.ch/pubs/files/deep-it-2019.pdf`

Two geometric notions of exponential expressivity

Partitions of the domain and path length

Prior to the approximation rate results from Telgarsky 15' and Yarotsky 16', there were qualitative geometric results showing showing potential for exponential expressivity:

- On the number of response regions of deep feedforward networks with piecewise linear activations (Pascanu et al. 14') `https://arxiv.org/pdf/1312.6098.pdf`
- On the expressive power of deep neural networks (Raghu et al. 16') `https://arxiv.org/abs/1606.05336`
- Trajectory growth lower bounds for random sparse deep ReLU networks (Price et al. 19') `https://arxiv.org/abs/1911.10651`

ReLU hyperplane arrangement

Partition of the input domain $\mathbb{R}^{n_0}$: one layer

The action of ReLU to an affine transform is a linearly increasing function orthogonal to hyperplanes; let $W \in \mathbb{R}^{n_1 \times n_0}$ then:

$$H_i := \{x \in \mathbb{R}^{n_0} : \; W_i x + b_i = 0\} \quad \forall i \in [n_1]$$

where W_i is the i^{th} row of W.
The normals to these hyperplanes partition the input dimension n_0, and if W is in general position (all subsets of rows are maximal rank), then the number of partitions is:

$$\sum_{j=0}^{n_0} \binom{n_1}{j}$$

https://arxiv.org/pdf/1312.6098.pdf

ReLU hyperplane arrangement

Partition of the input domain $\mathbb{R}^{n_0}$: with depth

Mathematical Institute

The number of partitions in one layer is lower bounded by

$$\sum_{j=0}^{n_0} \binom{n_1}{j} \geq n_1^{\min\{n_0, n_1/2\}}$$

and each hidden layers can further subdivide these regions:

Theorem (Pascanu et al. 14')

An L layer DNN with ReLU activation, input $\mathbb{R}^{n_0}$, and hidden layers of width $n_1, n_2, \ldots, n_L$ partitions the input space into at least

$$\Pi_{\ell=0}^{L} n_\ell^{\min\{n_0, n_\ell/2\}}$$

This shows an exponential dependence on depth L.
`https://arxiv.org/pdf/1312.6098.pdf`

ReLU hyperplane arrangement

Partition of the input domain $\mathbb{R}^{n_0}$: plot Pascanu et al. 14'

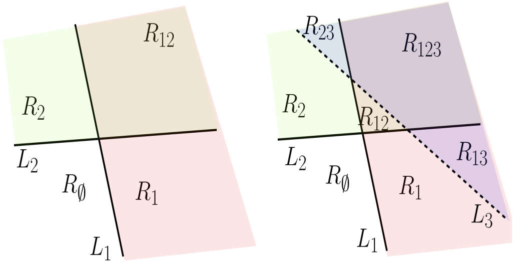
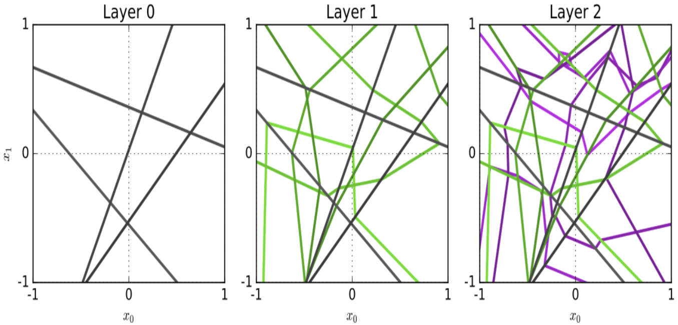


Figure 2: Induction step of the hyperplane sweep method for counting the regions of line arrangements in the plane.

`https://arxiv.org/pdf/1312.6098.pdf`

ReLU hyperplane arrangement

Partition of the input domain $\mathbb{R}^{n_0}$: plot Raghu et al. 16'



https://arxiv.org/abs/1606.05336

This "activation region" perspective is a useful intuition for ReLU, but lacks the quantitative convergence rates we observed in more recent approximation theory results of Yarotsky 16'.

Random initialisations and DNNs

DNNs are typically first trained from random values

A random network $f_{NN}(x; \mathcal{P}, \mathcal{Q})$ denotes a deep neural network:

$$h^{(d)} = W^{(d)} z^{(d)} + b^{(d)}, \qquad z^{(d+1)} = \phi(h^{d}), \qquad d = 0, \ldots, L-1,$$

which takes as input the vector x, and is parameterised by random weight matrices $W^{(d)}$ with entries sampled iid from the distribution $\mathcal{P}$, and bias vectors $b^{(d)}$ with entries drawn iid from distribution $\mathcal{Q}$.

While our goal is always to train a network, DNNs typically start as random networks which influence their ability to be trained.

Popular choices are Gaussian, $\mathcal{P} = \mathcal{N}(0, \sigma_w^2)$, or uniform, $\mathcal{P} = \mathcal{U}(-C_w, C_w)$ initialisations.
(*Note, for random networks we use $\phi(\cdot)$ as the nonlinear activation and σ to denote variance.)

Trajectory length of random DNNs

A geometric notion of expressivity

Raghu et al. 16' introduced the notion of trajectory length

$$l(x(t)) = \int_t \left|\left|\frac{dx(t)}{dt}\right|\right| dt.$$

as a measure of expressivity of a DNN. In particular, they considered passing a simple geometric object $x(t)$, such as a line $x(t) = tx_0 + (1-t)x_1$ for $x_0, x_1 \in \mathbb{R}^k$ and measure the expected length of the output of the random DNN at layer d:

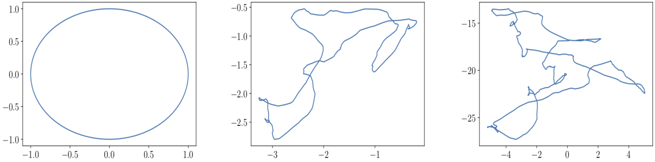
$$\frac{\mathcal{E}\left[\ell(z^{(d)})\right]}{\ell(x(t))}$$

`https://arxiv.org/abs/1606.05336`

Example of circle passed through a random DNN

Complexity of output increasing with depth

A circle passed through a random DNN and the pre-activation output $h^{(d)}$ at layers $d = 6$ and 12.



(a) Input (b) Layer 6 (c) Layer 12

Figure 1: A circular trajectory, passed through a ReLU network with $\sigma_w = 2$. The plots show the pre-activation trajectory at different layers projected down onto 2 dimensions.

DNNs can be used to *generative* data, GANs, and there one might consider the complexity of the manifold the GAN can generate as a measure of expressivity.

Random DNN: expected path length lower bound

Path length bound dependence on σ_w.

Consider random DNNs of width n and depth L with weights and bias are drawn i.i.d. $W^{(\ell)}(i,j) \sim \mathcal{N}(0, \sigma_w^2/n)$, $b^{(\ell)}(j) \sim \mathcal{N}(0, \sigma_b^2)$

Theorem (Raghu et al. 16')

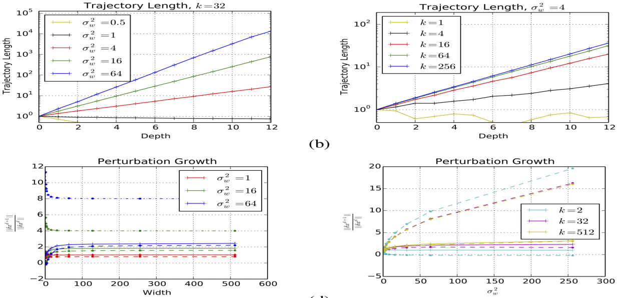
Consider as input a one dimensional trajectory $x(t)$ with arc-length $\ell(x(t)) = \int_t \left\| \frac{dx(t)}{dt} \right\| dt$ and let $z^{(L)}(t)$ be the output of the Gaussian random feedforward network with ReLu activations, then

$$\frac{\mathcal{E}\left[\ell(z^{(L)})\right]}{\ell(x(t))} \geq \mathcal{O}\left(\left(\frac{\sigma_w}{(\sigma_w^2+\sigma_b^2)^{1/4}} \cdot \frac{n^{1/2}}{(n+(\sigma_w^2+\sigma_b^2)^{1/2})^{1/2}}\right)^L\right).$$

https://arxiv.org/abs/1606.05336

Exponential growth of path length with depth.

Empirical experiments for htanh activation



(a) (b) (c) (d)

Figure 2: The exponential growth of trajectory length with depth, in a random deep network with hard-tanh nonlinearities. A circular trajectory is chosen between two random vectors. The image of that trajectory is taken at each layer of the network, and its length measured. *(a,b)* The trajectory length vs. layer, in terms of the network width k and weight variance σ_w^2, both of which determine its growth rate. *(c,d)* The average ratio of a trajectory's length in layer $d+1$ relative to its length in layer d. The solid line shows simulated data, while the dashed lines show upper and lower bounds (Theorem 1). Growth rate is a function of layer width k, and weight variance σ_w^2.

`https://arxiv.org/pdf/1611.08083.pdf`

Random DNN: expected path length lower bound

Generalised and simplified lower bound

Theorem (Price et al. 19')

Let $f_{NN}(x; \alpha, \mathcal{P}, \mathcal{Q})$ be a random sparse net with layers of width n. Then, if $\mathbb{E}[|u^T w_i|] \geq M\|u\|$, where w_i is the i^{th} row of $W \in \mathcal{P}$, and u and M are constants, then

$$\mathbb{E}[l(z^{(L)}(t))] \geq \left(\frac{M}{2}\right)^L \cdot l(x(t))$$

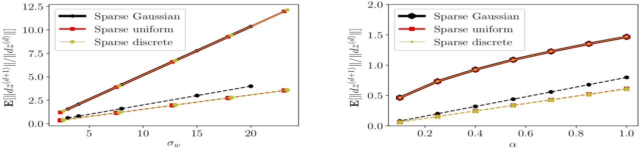
for $x(t)$ a 1-dimensional trajectory in input space.

Exponential growth with depth for random initialisations such as Gaussian, uniform, and discrete; e.g. for Gaussian $M = \sigma_w \sqrt{2/\pi}$.
`https://arxiv.org/abs/1911.10651`

Observed growth rate (solid) and bounds (dashed)

Empirical experiments showing dependence on σ_W and sparsity α

Price et al. 19' also extended the results to have all but α fraction of the entries in W equal to 0.



(a)

(b)

Figure 3: Expected growth factor, that is, the expected ratio of the length of any very small line segment in layer $d + 1$ to its length in layer d. Figure 3a shows the dependence on the variance of the weights' distribution, and Figure 3b shows the dependence on sparsity.

Unless σ_W or α small enough at initialisation the pre-activation output is exponentially complex.

`https://arxiv.org/abs/1911.10651`