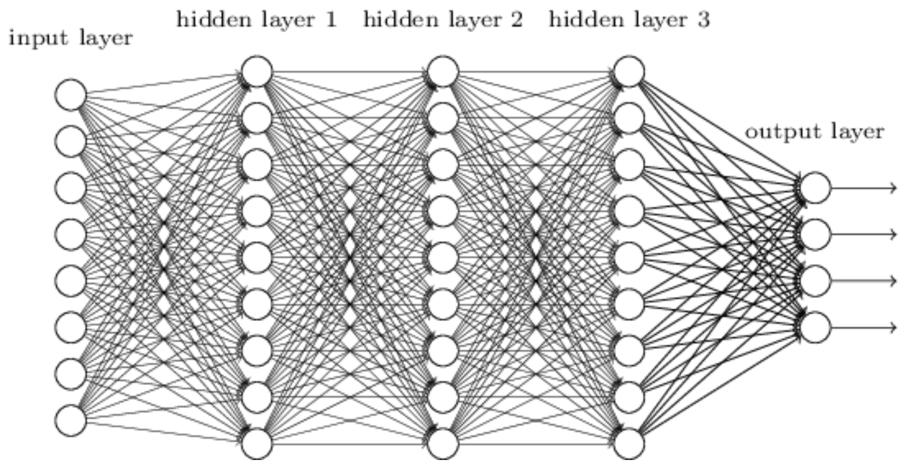
Machine learning with neural networks: the importance of data structure

Marc Mézard
Ecole normale supérieure - PSL University

IPAM Workshop, November 21, 2019
Los Angeles



Why does it work?

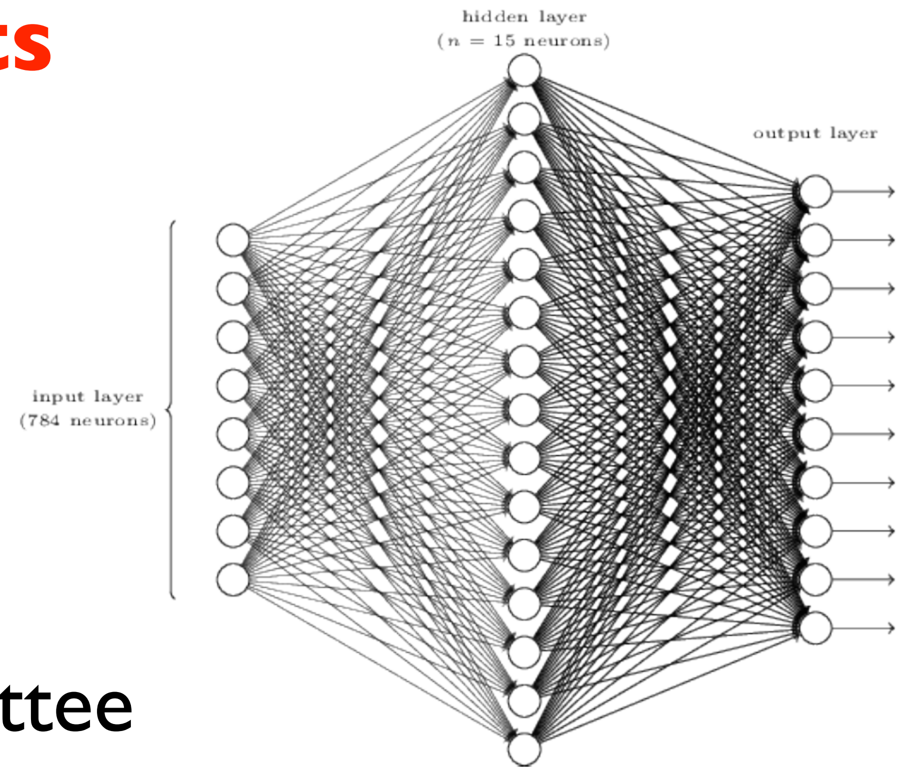
Data structure

- Hidden manifolds and sub manifolds
- Combinatorial structure
- Euclidean correlations

- Analyse data
- Build generative models that can be analyzed fully in some large size limit
- Understand mechanisms

S. Goldt, F. Krzakala, MM, L. Zdeborova
arXiv:1909.11500

Theory: Ensembles of data, ensemble of weights



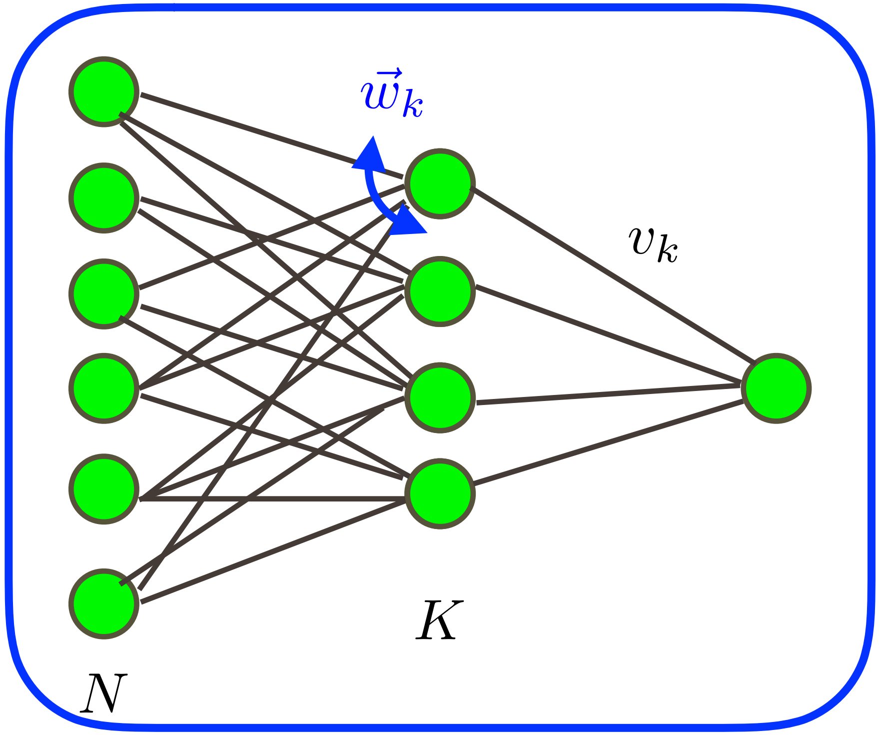
Mostly used so far
Data = input patterns
with **iid entries**

Perceptron learning, committee machine, teacher-student

Pattern μ , input entry i : $X_{\mu i} = \mathcal{N}(0,1)$ $P \times N$ matrix

NB Physicists use P patterns in N dimensions, statisticians use n patterns in p dimensions… **Sorry**

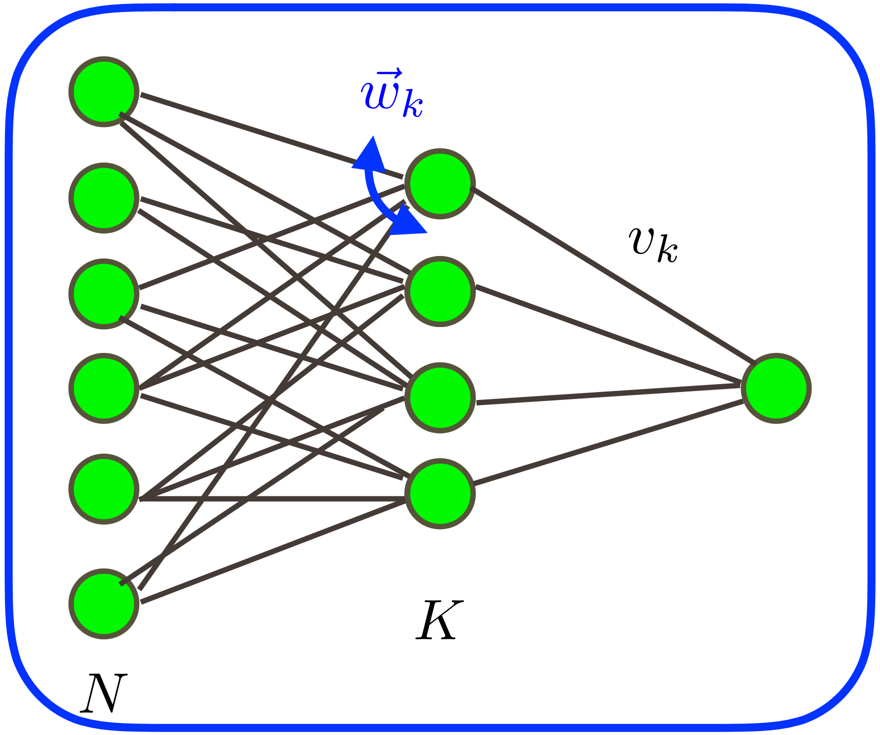
Differences between MNIST and iid data



Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_{k}^{K} v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right)$$

Differences between MNIST and iid data



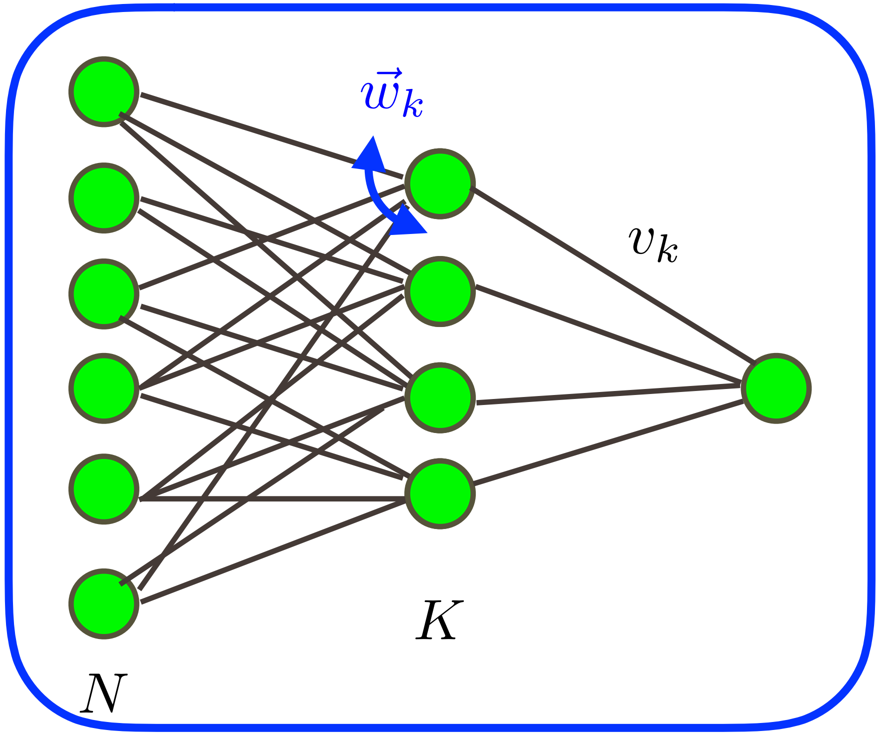
Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_{k}^{K} v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right)$$

Task 1: distinguish odd from even numbers in MNIST

$\phi_t(\vec{X}) = 1$ for even digits $\quad \phi_t(\vec{X}) = -1$ for odd digits

Differences between MNIST and iid data



Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_{k}^{K} v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right)$$

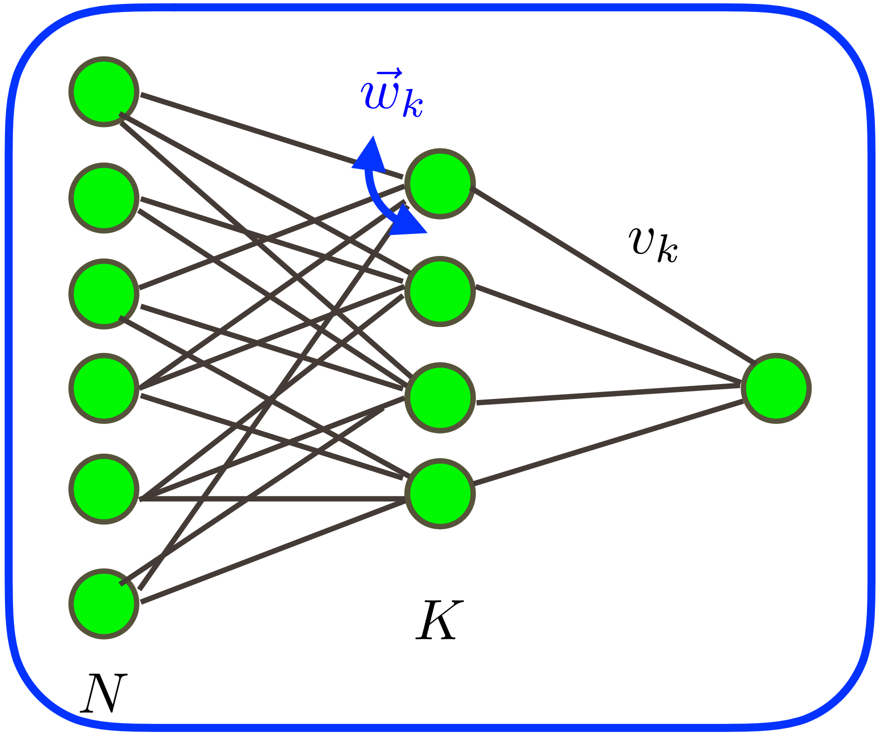
Task 1: distinguish odd from even numbers in MNIST

$\phi_t(\vec{X}) = 1$ for even digits $\quad \phi_t(\vec{X}) = -1$ for odd digits

Task 2: iid input data; desired output given by a 2-layer « teacher network » with M hidden units

$$\phi_t(\vec{X}) = \mathrm{Sign}\left[\sum_{m=1}^{M} \nu_m \, g\left(\vec{\omega}_m.\vec{X}/\sqrt{N}\right)\right]$$

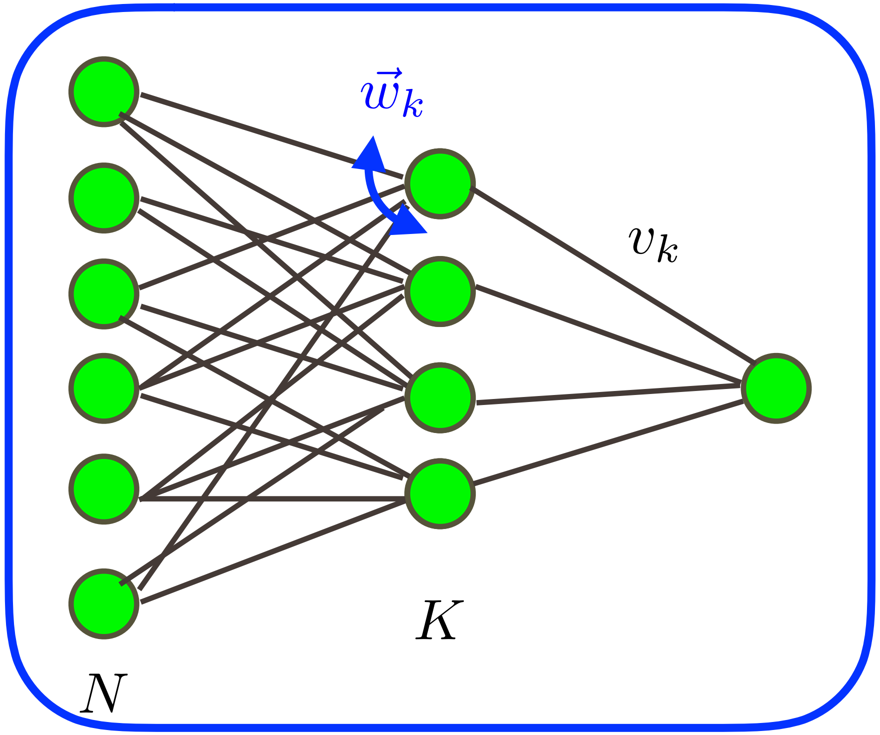
Differences between MNIST and iid data



Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_{k}^{K} v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right),$$

Differences between MNIST and iid data



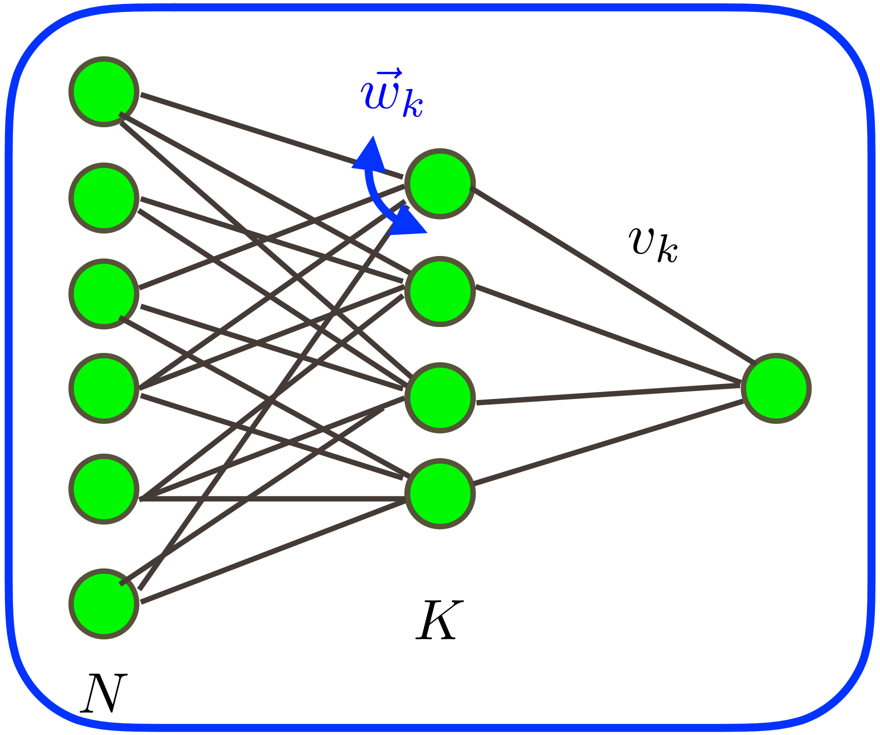
Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_{k}^{K} v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right),$$

Training error $\varepsilon_g = \frac{1}{2P}\sum_{\mu=1}^{P} \theta\left[\phi(\vec{X}_\mu) - \phi_t(\vec{X}_\mu)\right]^2$

Generalization error: same with P^* new patterns

Differences between MNIST and iid data



Learn using a 2-layer neural net, K hidden units

$$\phi(\vec{X}) = \sum_k^K v_k g\left(\vec{w}_k.\vec{X}/\sqrt{N}\right),$$

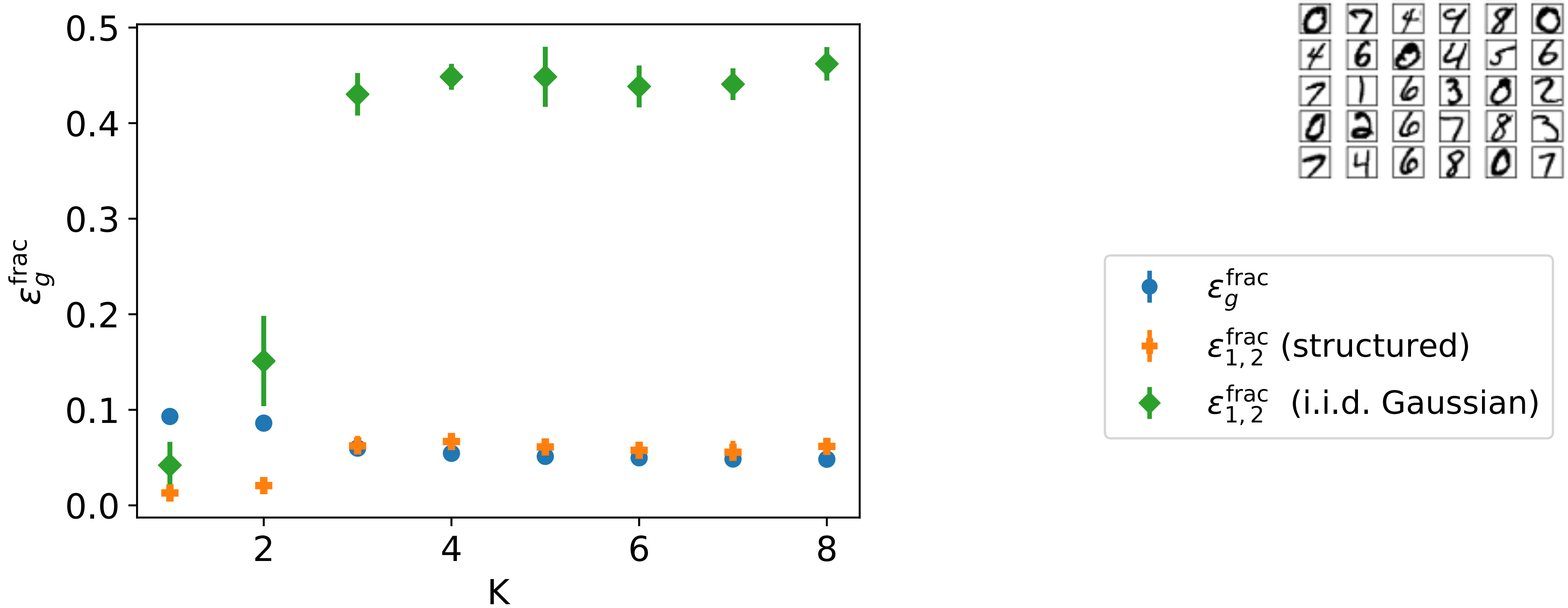
Training error $$\varepsilon_g = \frac{1}{2P}\sum_{\mu=1}^{P}\theta\left[\phi(\vec{X}_\mu) - \phi_t(\vec{X}_\mu)\right]^2$$

Generalization error: same with P^* new patterns

Also monitored: difference between two learning trials with different initial conditions

$$\varepsilon_{12} = \frac{1}{2P}\sum_{\mu=1}^{P}\theta\left[\phi_1(\vec{X}_\mu) - \phi_2(\vec{X}_\mu)\right]^2$$

MNIST data



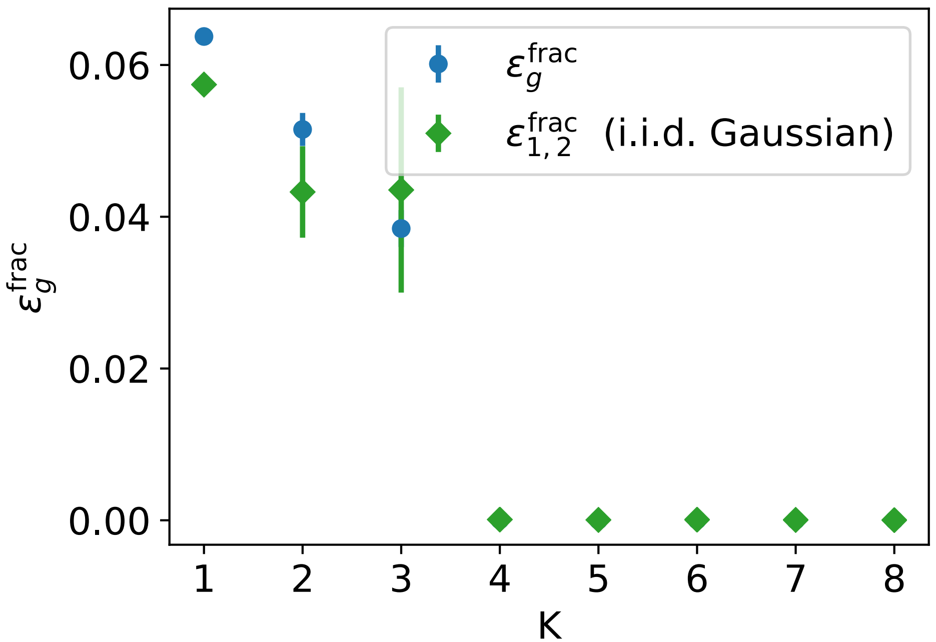
Generalization error decreases with K

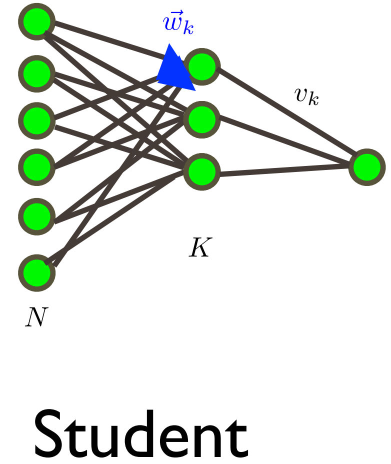
The difference on MNIST between two trials agrees with generalization error

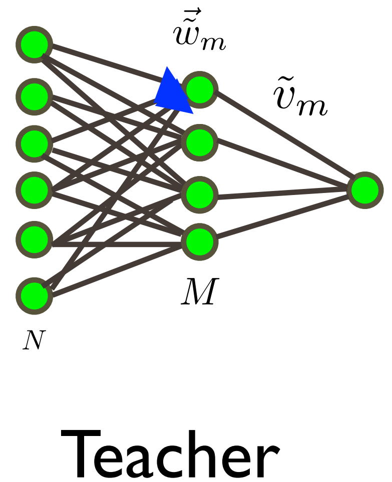
The difference on random images between two trials is large (nearly uncorrelated functions)

iid data and teacher network

$M = 4$



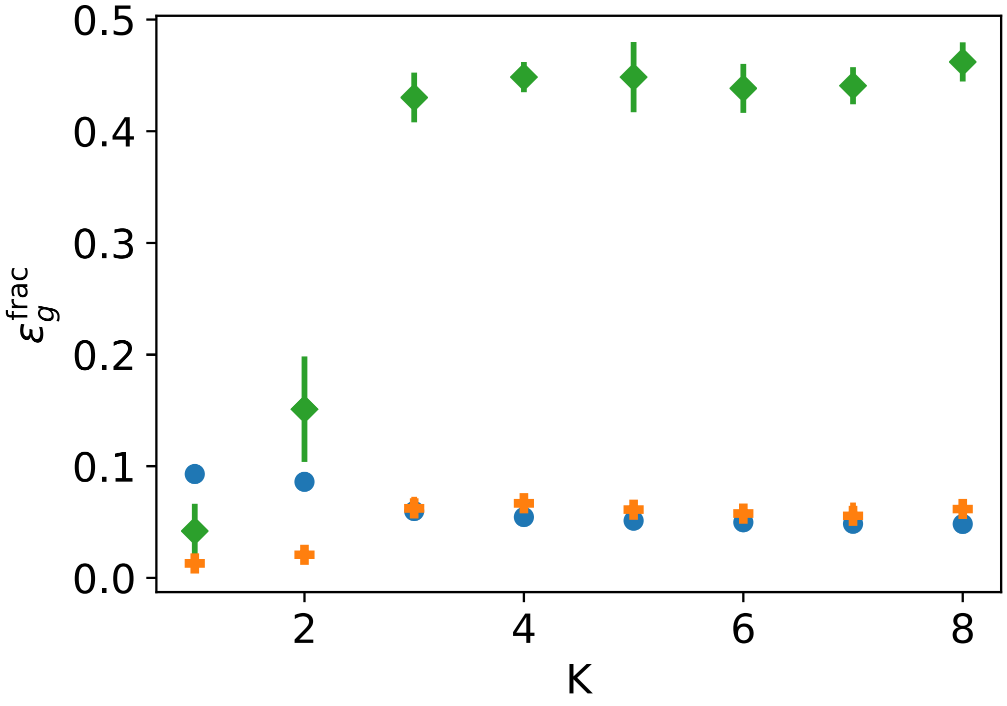


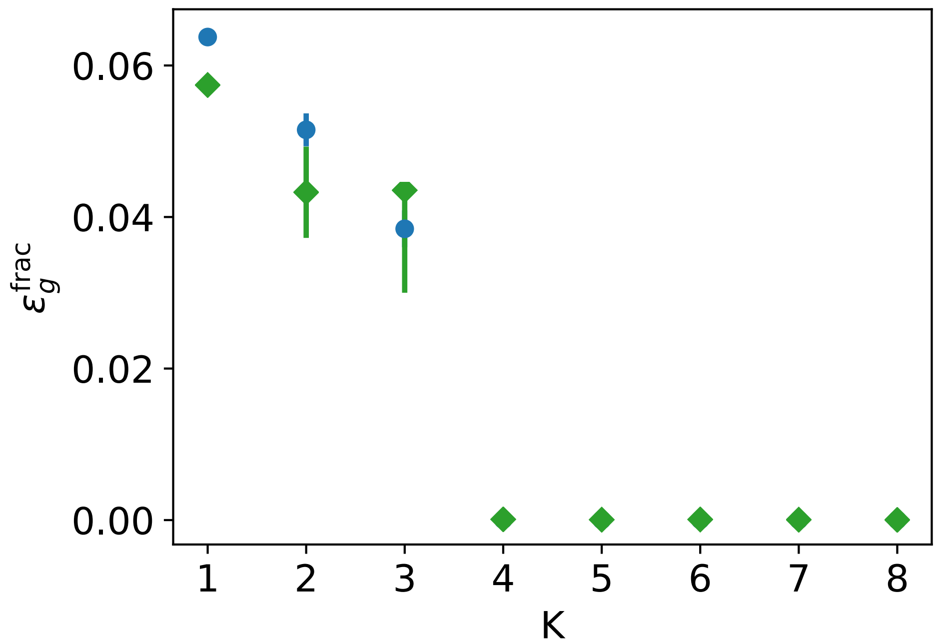


Generalization error decreases with K, vanishes for $K \geq M$

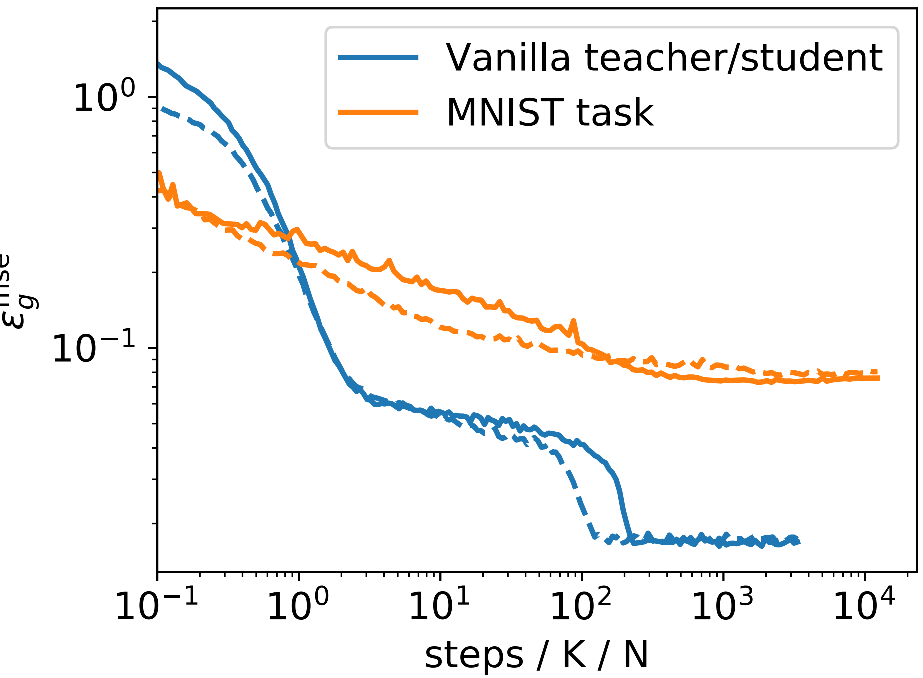
The difference on random images between two trials is equal to ε_g. For $K \geq M$ the two trials learn the same global function

MNIST versus iid data





Learning dynamics

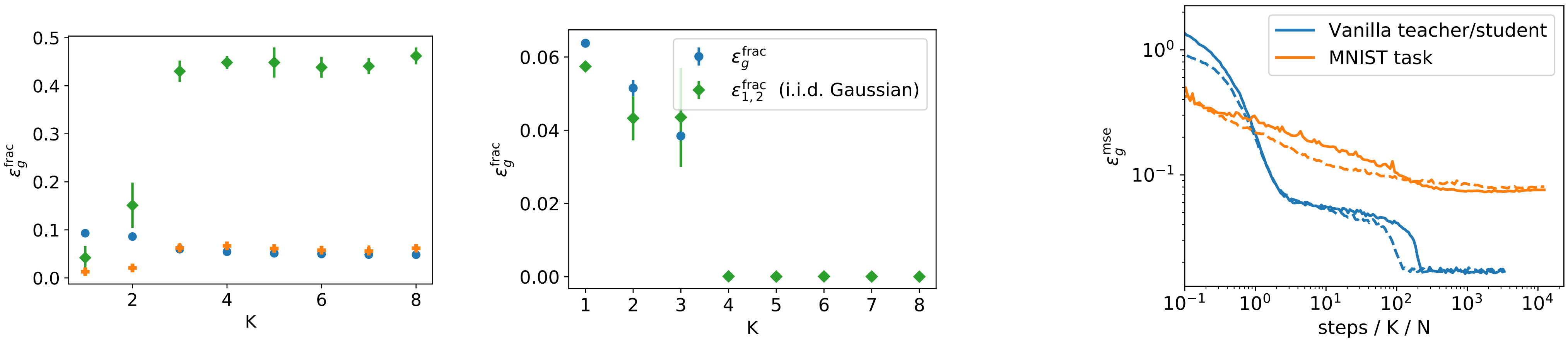


Plateau in the « teacher-student » setup

M. Biehl and H Schwarze 95, Saad and Solla 95

After some time the dynamics stabilize in a metastable state where all the hidden units have roughly the same overlap with all the teacher vector. Long plateau before the specialization of hidden units occurs.

Differences between MNIST and iid data

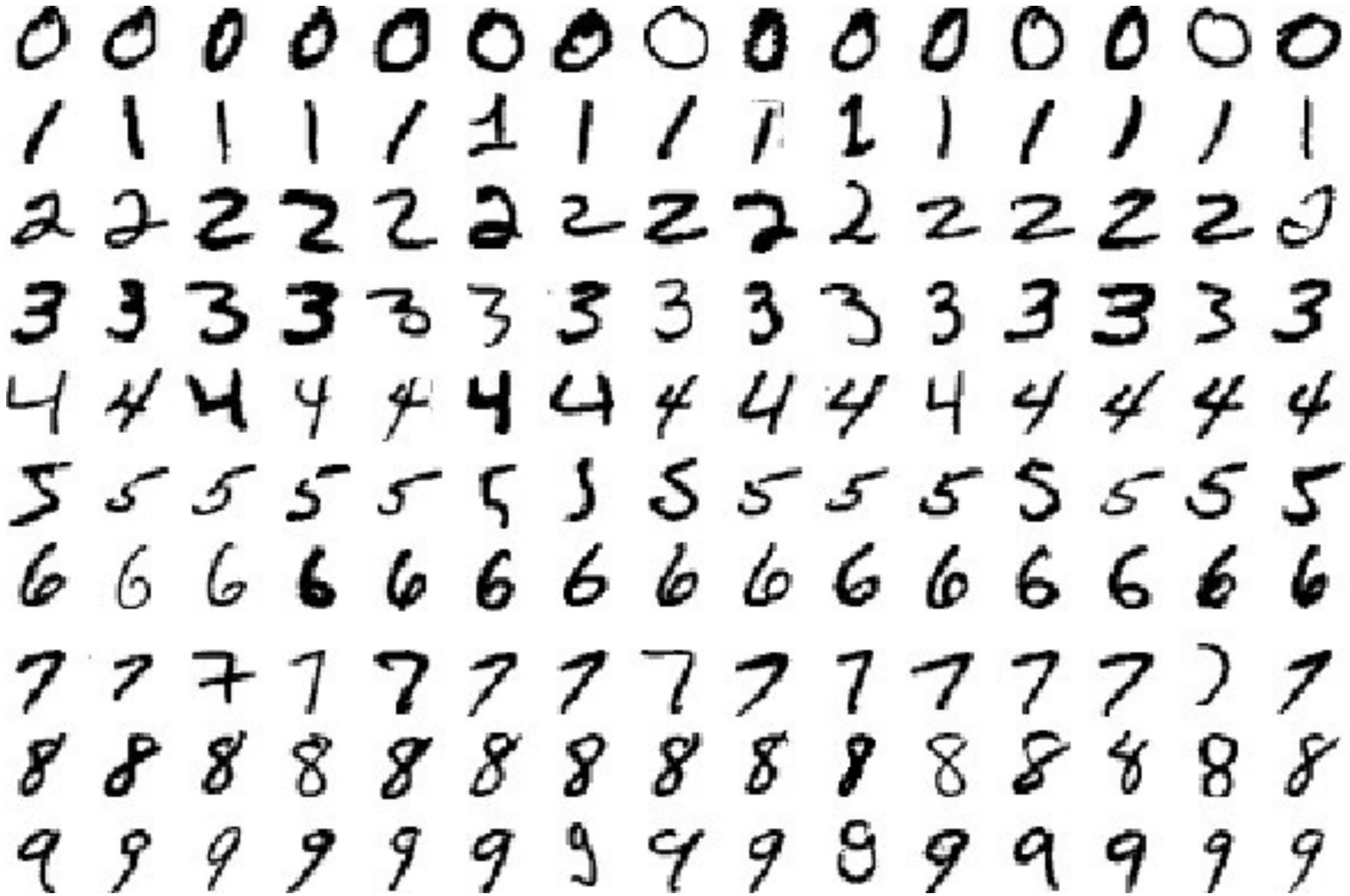


Two different trials learn the same function in iid data teacher-student, completely different functions in MNIST (outside of the hidden manifold)

Plateau in the learning of teacher-student with iid data, not seen in MNIST

The hidden manifold of data

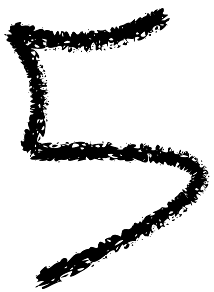
MNIST



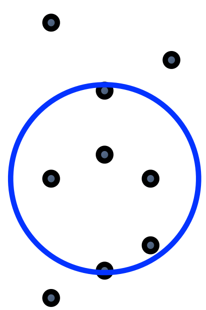
Input space: dimension $28^2 = 784$

The hidden manifold of data

Input space: dimension $28^2 = 784$



Manifold of handwritten digits in MNIST:



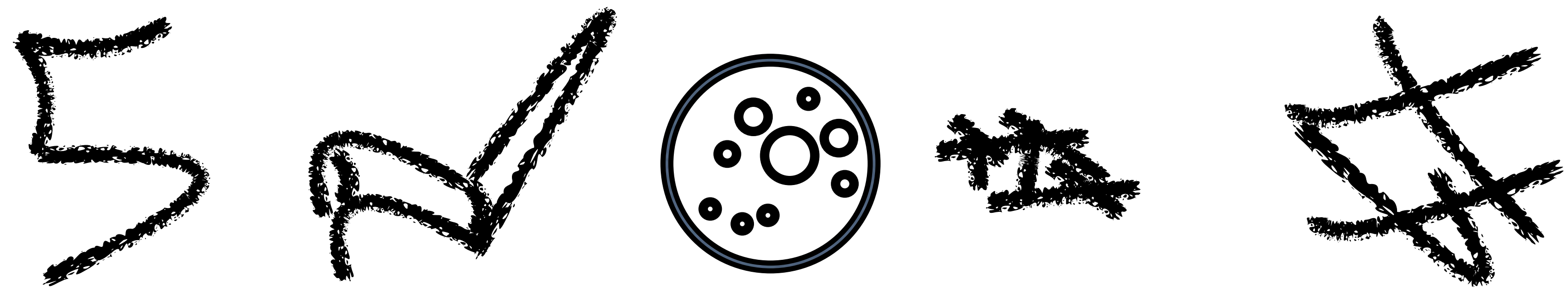
Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$

$p \simeq cR^d$

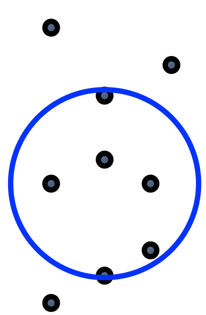
Grassberger Procaccia 83, Costa Hero 05, Heinz Audibert 05, Ansuini et al. 19, Spigler et al. 19…

The hidden manifold of data

Input space: dimension $28^2 = 784$



Manifold of handwritten digits in MNIST:

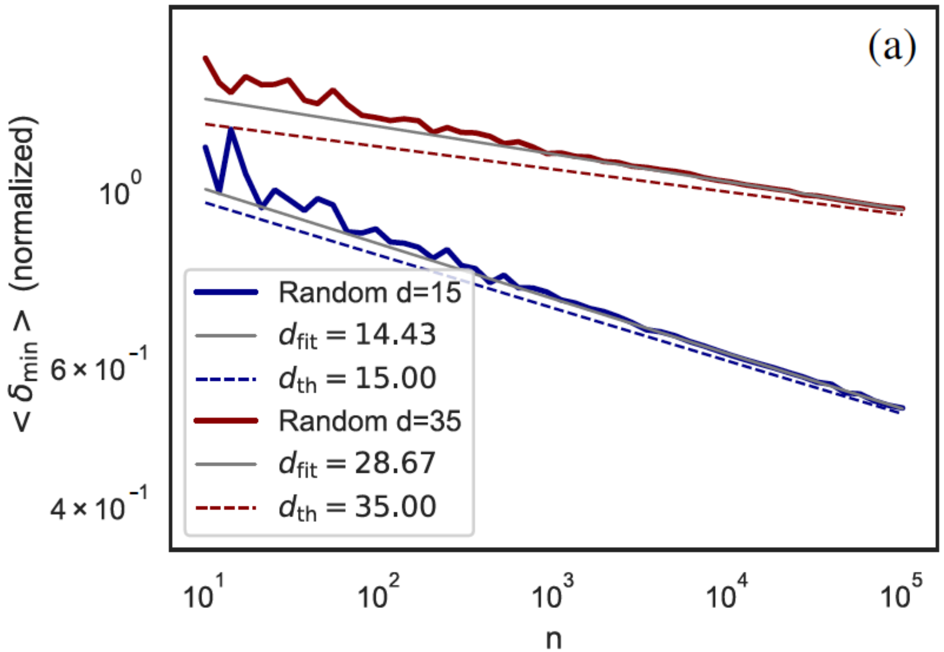


$p \simeq cR^d$

Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$

Grassberger Procaccia 83, Costa Hero 05, Heinz Audibert 05, Ansuini et al. 19, Spigler et al. 19…

The hidden manifold of data



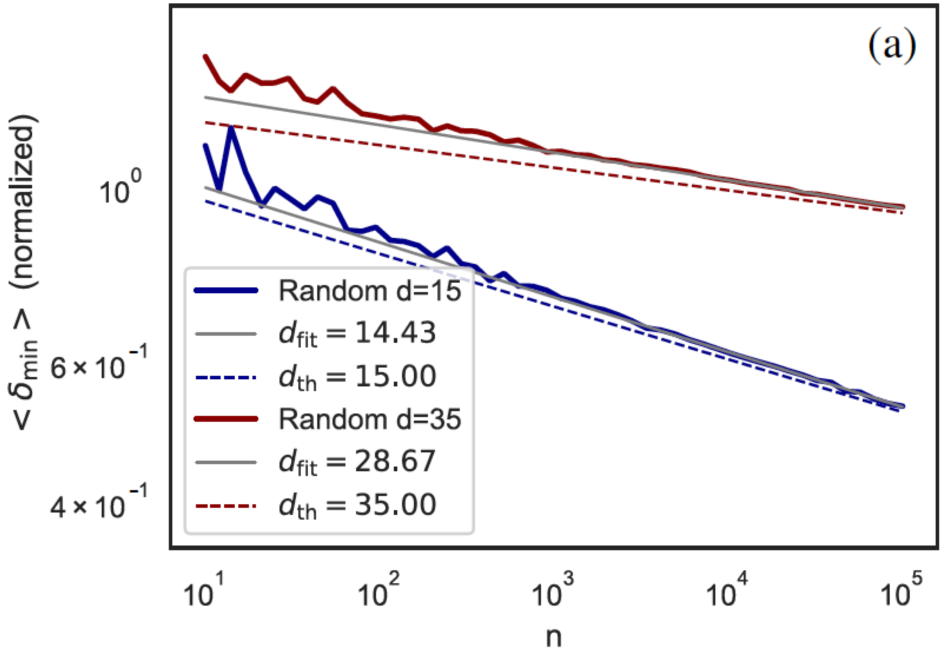
MNIST: $d = 784$

$d_{\text{eff}} \simeq 15$

Spigler et al. 19

Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$

The hidden manifold of data

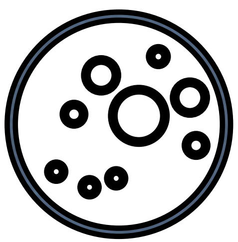


MNIST: $d = 784$

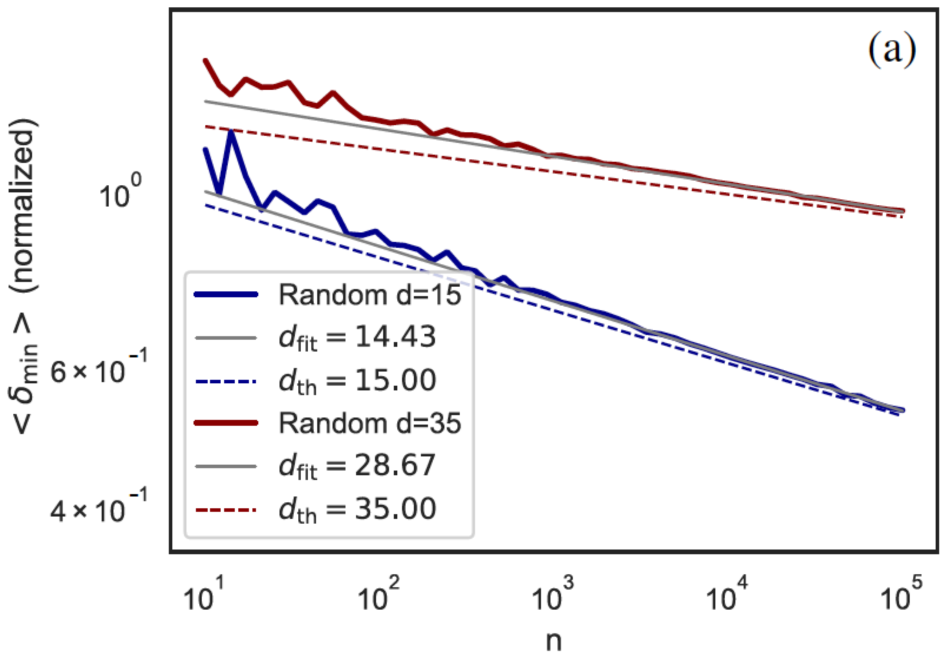
$d_{\text{eff}} \simeq 15$

Spigler et al. 19

Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$



The hidden manifold of data

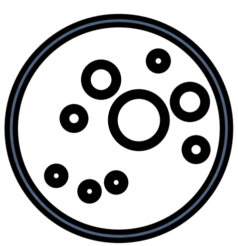


MNIST: $d = 784$

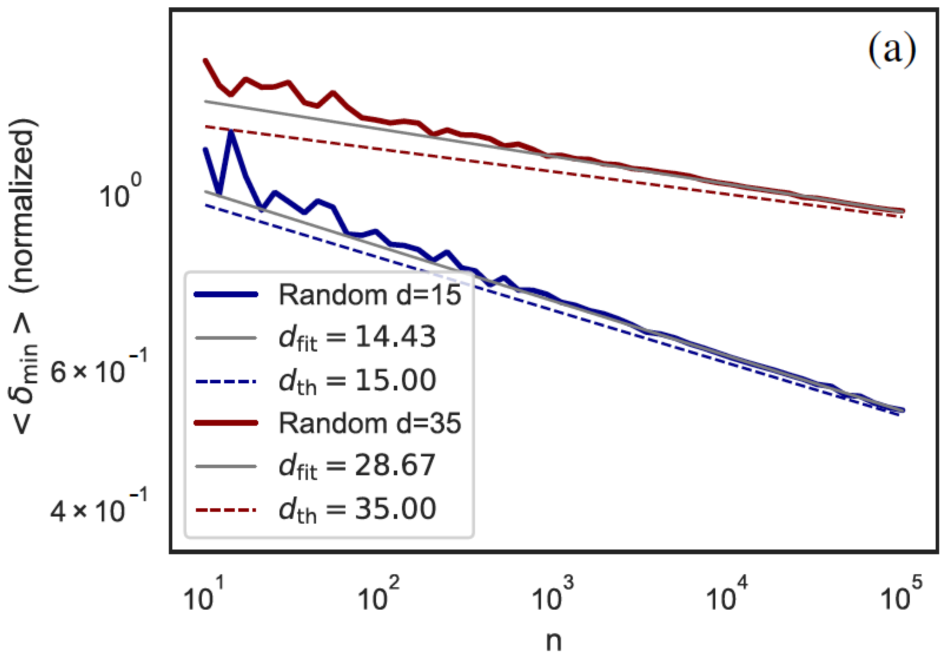
$d_{\mathrm{eff}} \simeq 15$

Spigler et al. 19

Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$



The hidden manifold of data

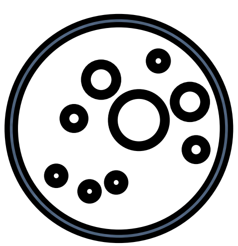


MNIST: $d = 784$

$d_{\rm eff} \simeq 15$

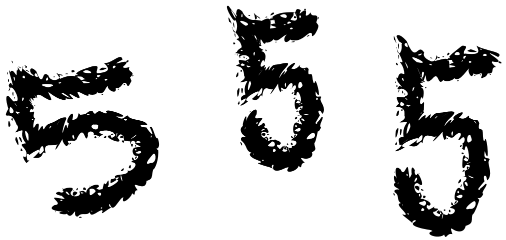
Spigler et al. 19

Nearest neighbors' distance : $R_{nn} \simeq p^{-1/d}$



The neural net should answer: this image does not seem to be a handwritten digit

Structure of the task: perceptual sub-manifolds



$$d_{\mathrm{eff}}(5) \simeq 12$$

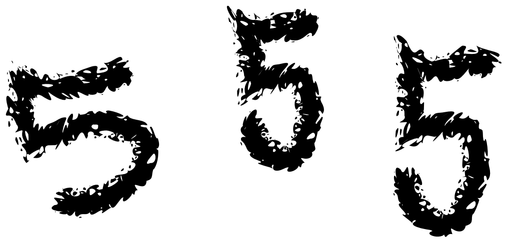
Hein Audibert 05

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

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

MNIST problem: in the **15-dim manifold** of handwritten digits, identify the **10 perceptual sub manifolds** associated with each digit, of **dimensions between 7 and 13**…

Structure of the task: perceptual sub-manifolds



$d_{\text{eff}}(5) \simeq 12$

Hein Audibert 05

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

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

MNIST problem: in the **15-dim manifold** of handwritten digits, identify the **10 perceptual sub manifolds** associated with each digit, of **dimensions between 7 and 13**…

… from an input in 784 dimensions!

A new ensemble for the hidden manifold and for the task to be achieved

S. Goldt, F. Krzakala MM L. Zdeborova

arXiv:1909.11500

An ensemble for the hidden manifold

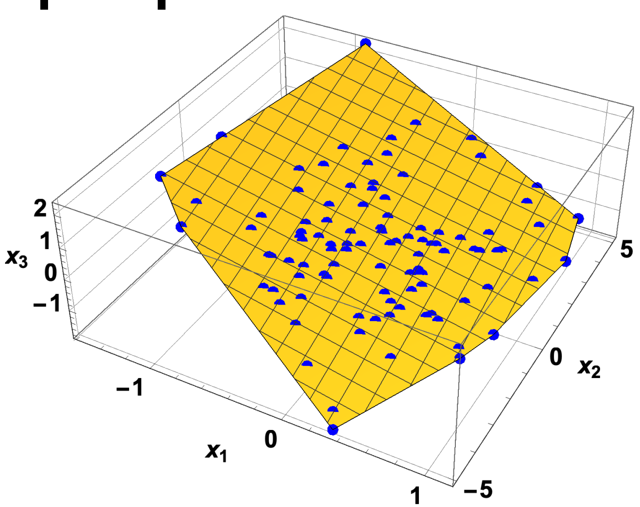
Pattern μ:
$$X_{\mu i} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_{\mu r} F_{ir}\right]$$

Data = input patterns built from R features $\vec{F}_r$

A feature is a N component vector in the input space

Each pattern is built from a weighted superposition of features (feature r has weight C_r):

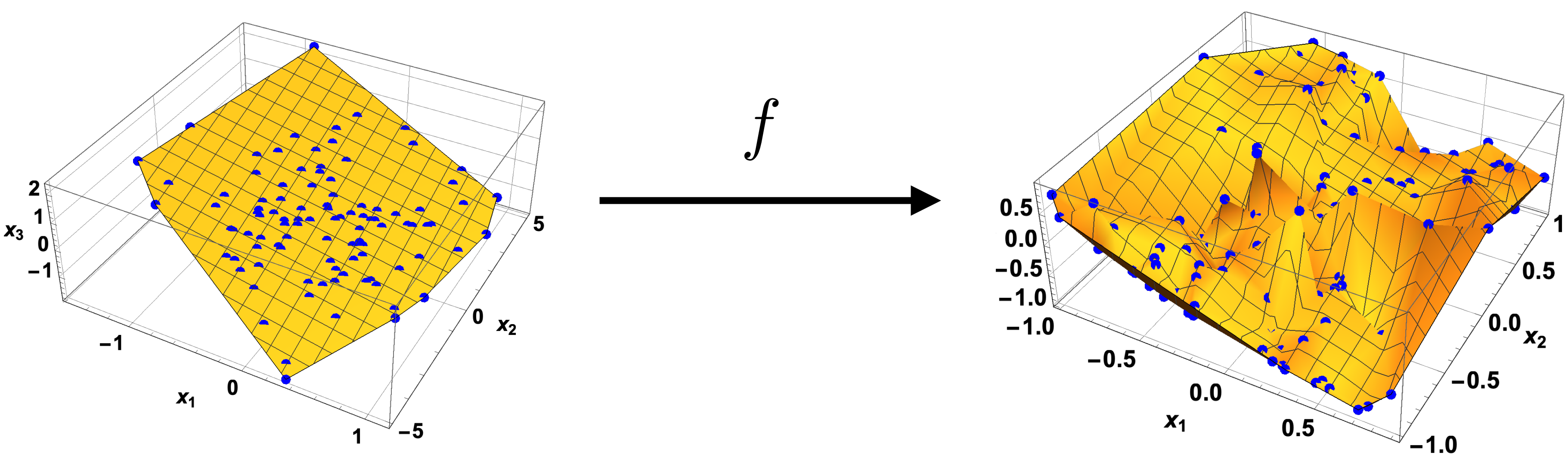
$$\sum_{r=1}^{R} C_r \vec{F}_r$$



An ensemble for the hidden manifold

$$X_{\mu i} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_{\mu r} F_{ir}\right]$$

The R-dimensional data manifold is folded by applying the non-linear function f



An ensemble for the task

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

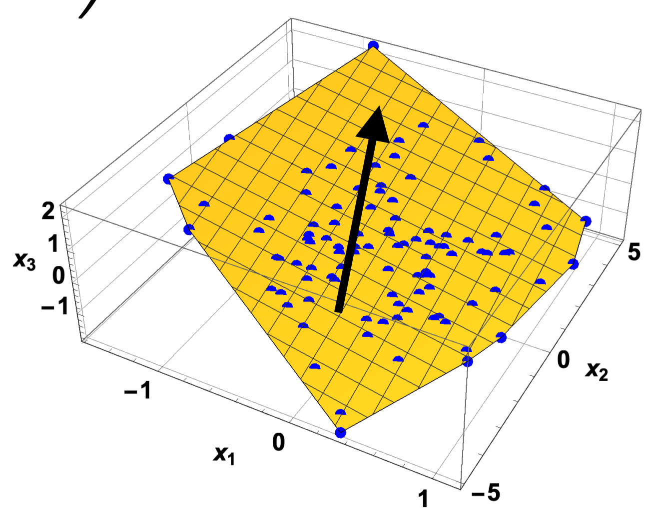
« Latent representation »: $\{C_r\}$

iid

Desired output = **function of latent representation**

Examples: $y = g\left(\sum_{r=1}^{R} \tilde{w}_r C_r\right)$

(perceptron in hidden manifold)



An ensemble for the task

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

« Latent representation »: $\{C_r\}$

Desired output (task) = function of latent representation

Examples: $$y = g\left(\sum_{r=1}^{R} \tilde{w}_r C_r\right)$$ (perceptron in latent space)

$$y = \sum_{m=1}^{M} \tilde{v}_m \; g\left(\sum_{r=1}^{R} \tilde{w}_{mr} C_r\right)$$ (2 layers nn in latent space)

Manifold of data and sub manifolds of the task

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

« Latent representation »: $\{C_r\}$

Hidden manifold of data: folded R-dimensional manifold

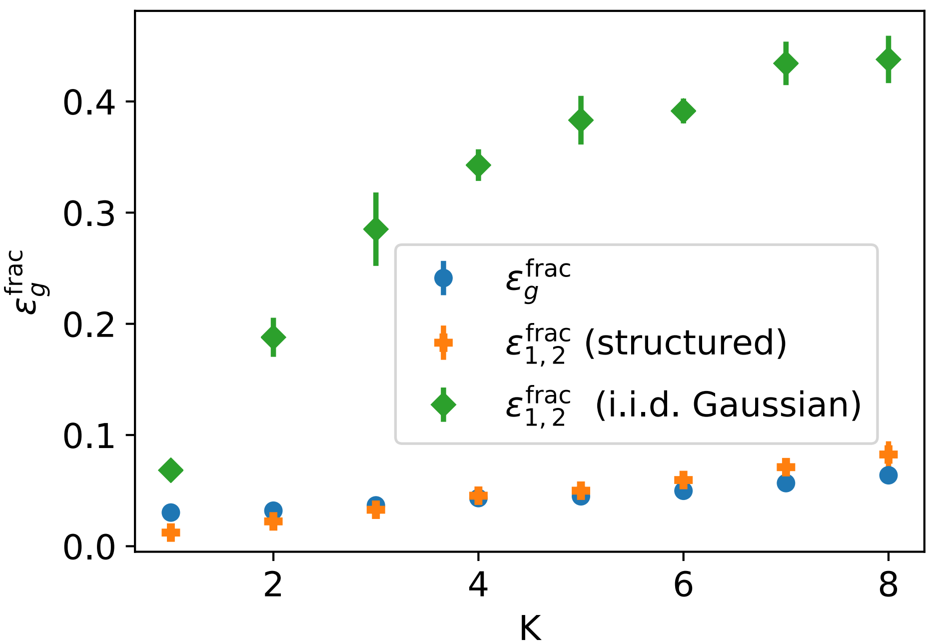
Task $\quad y = \sum_{m=1}^{M} \tilde{v}_m \, g\left(\sum_{r=1}^{R} \tilde{w}_{mr} C_r\right)$

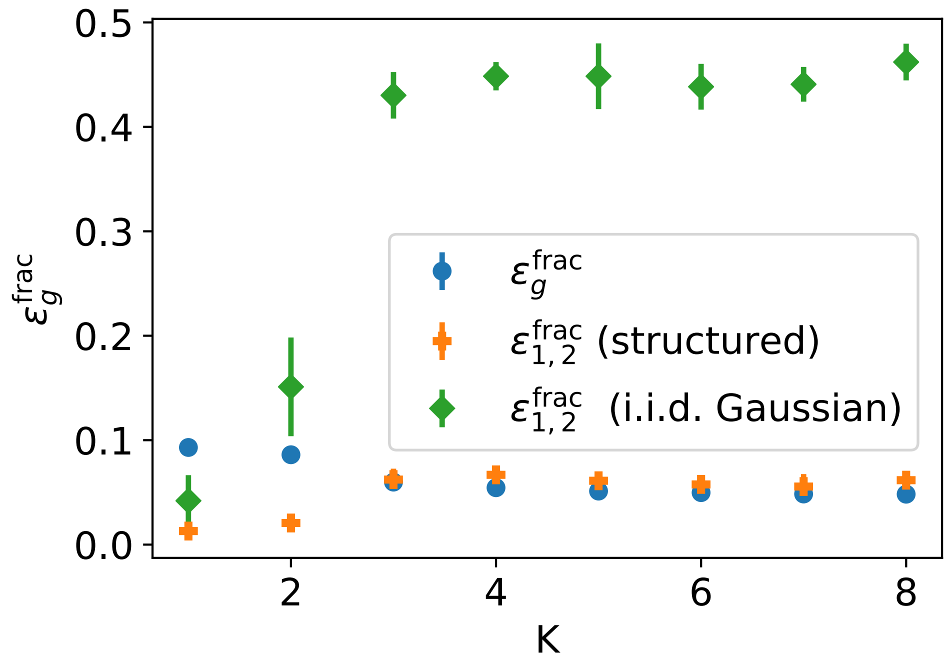
depends on $\{\tilde{w}_m . C\}, \quad m \in \{1, ...M\}$

where $\{\tilde{w}_m\}$ and C live in a R-dim space

For $M < C$ perceptual sub manifold = moving in directions orthogonal to the $\{\tilde{w}_m\}$, in latent space

Experimenting with the « hidden manifold model »



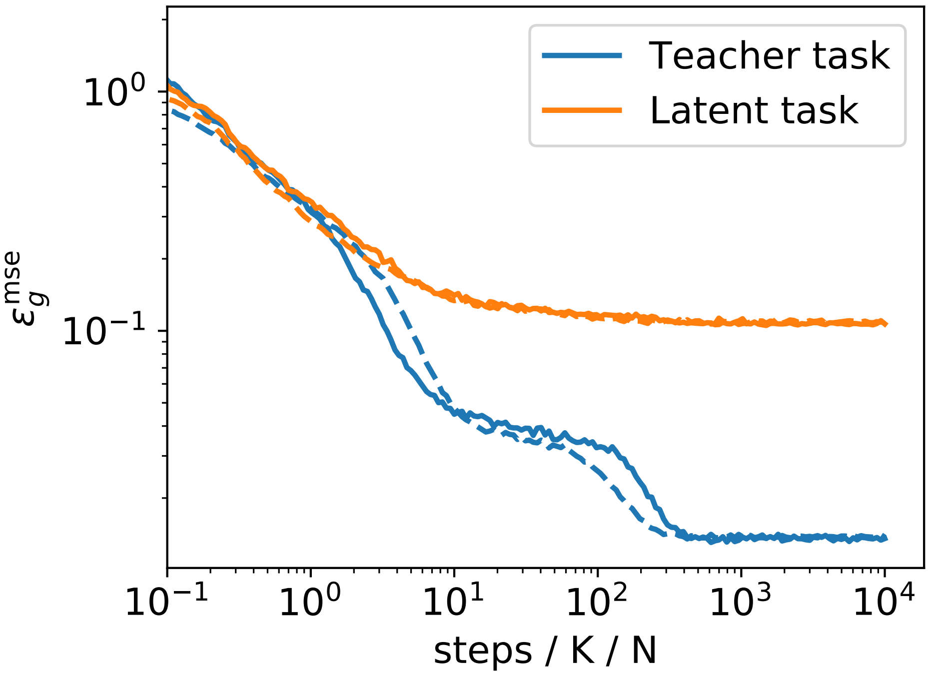


Hidden manifold model

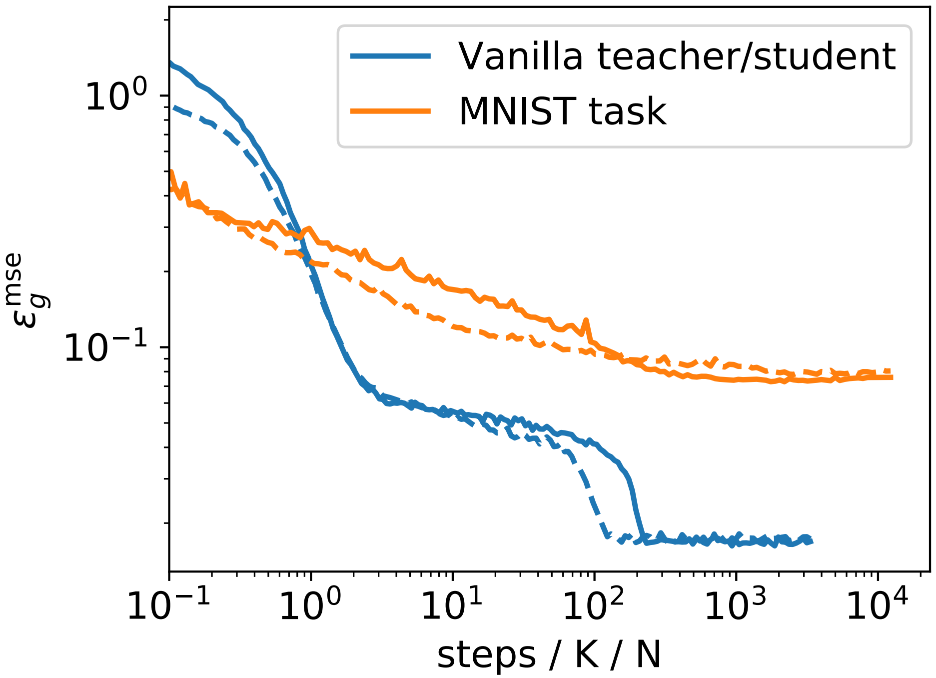
$R = 10$

MNIST

Experimenting with the « hidden manifold model »



Hidden manifold model



MNIST

Hidden manifold model

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Data. « Latent representation »: $\{C_r\}$

Desired output (task) = function of latent representation

Example $$y = g\left(\sum_{r=1}^{R} \tilde{w}_r C_r\right)$$

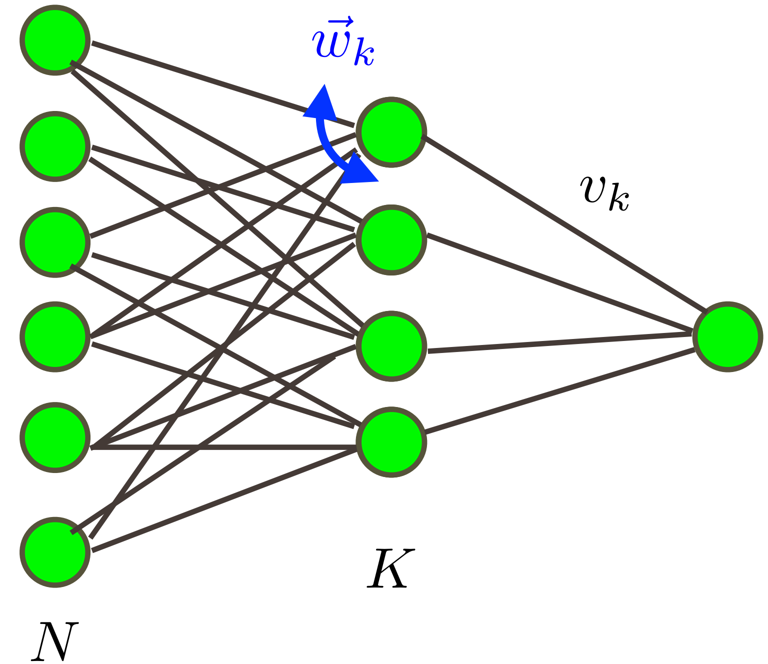
- Does not have the pathologies of teacher-student setup with iid data
- Learning and generalization phenomenology $\sim$ MNIST
- Can be studied analytically

Analytic study of the hidden manifold model

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



Solvable limit = thermodynamic limit with extensive latent dimension $N \to \infty$, $R \to \infty$, $P \to \infty$

With fixed $R/N = \gamma\,,\quad P/N = \alpha\,,\quad K$

Analytic study of the hidden manifold model

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

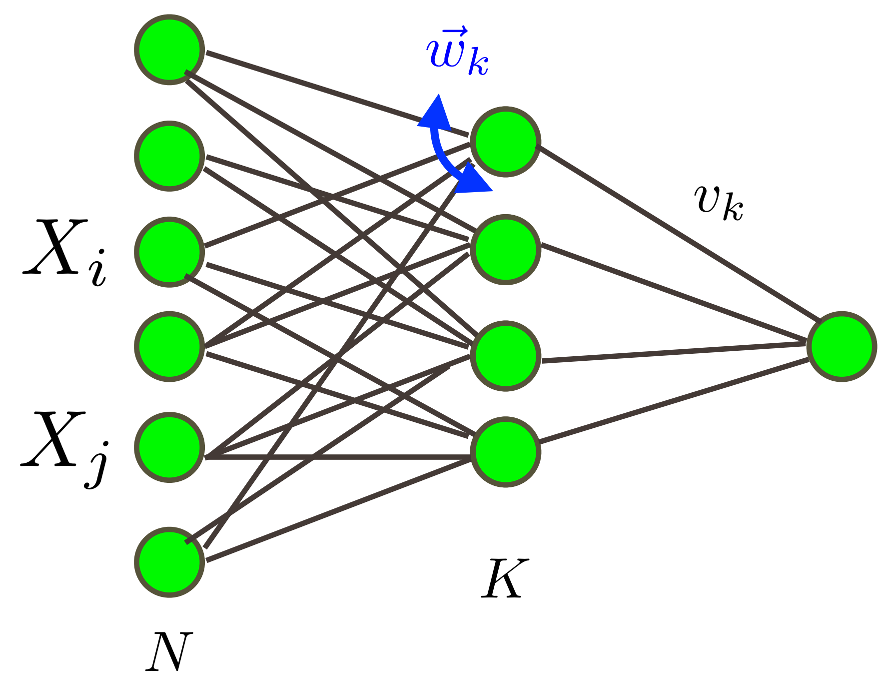
iid

balanced:

$$F_{ri} = O(1)$$

$$\frac{1}{N}\sum_i F_{ri}F_{si} = O(1/\sqrt{N})$$

$$\frac{1}{N}\sum_i F_{ri}F_{ri} = 1$$

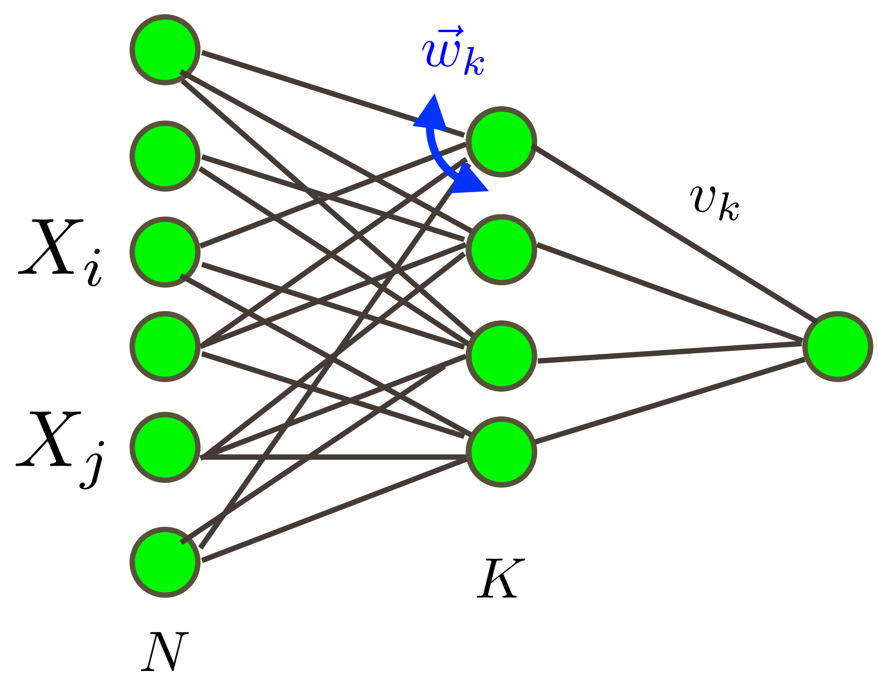


Analytic study of the hidden manifold model

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



$$X_i = f[u_i]$$

$$u_i = \frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r F_{ri}$$

Gaussian, weakly correlated $O(1/\sqrt{N})$
when F_{ri} are balanced and $O(1)$

$$\mathbb{E}\left(f[u_i]f[u_j]\right) = \langle f(u)\rangle^2 + \langle u f(u)\rangle^2 \mathbb{E}\left(u_i u_j\right)$$

u Gaussian $\mathcal{N}(0,1)$

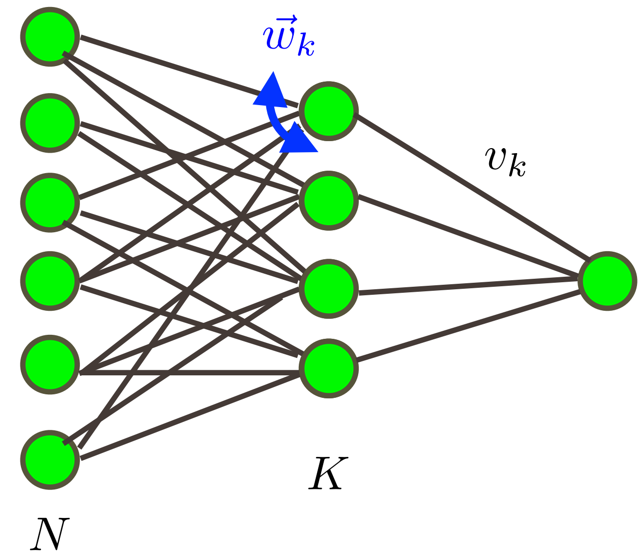
Gaussian Equivalence Theorem (GET)

$$u_i = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} \boxed{C_r} F_{ri}$$

iid

$$X_i = f[u_i]$$

Inputs of hidden units: $$\lambda^k = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k f[u_i]$$



GET: In the thermodynamic limit, the variables λ^k have a Gaussian distribution, with covariance

$$\mathbb{E}[\tilde{\lambda}^k \tilde{\lambda}^\ell] = (c - a^2 - b^2) W^{k\ell} + b^2 \Sigma^{k\ell}$$

$$W^{k\ell} \equiv \frac{1}{N} \sum_{i=1}^{N} w_i^k w_i^\ell \qquad \Sigma^{k\ell} \equiv \frac{1}{R} \sum_{r=1}^{R} S_r^k S_r^\ell \qquad S_r^k \equiv \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k F_{ir}$$

$c = \langle f(u)^2 \rangle \qquad a = \langle f(u) \rangle \qquad b = \langle u f(u) \rangle \qquad u$ Gaussian $\mathcal{N}(0,1)$

Gaussian Equivalence Theorem (GET)

$$u_i = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} C_r F_{ri}$$

$$X_i = f[u_i]$$

Inputs of hidden units:

$$\lambda^k = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k f[u_i]$$

GET in a nutshell: in the thermodynamic limit (with extensive latent dimension of the hidden manifold, $R = \gamma N$), the inputs of hidden units have Gaussian distribution. Then the model is solvable.

NB: F_{ri} and w_i^k are not necessarily random, but balanced

$$S^{k_1 k_2 \ldots k_p}_{r_1 r_2 \ldots r_q} = \frac{1}{\sqrt{N}} \sum_i w_i^{k_1} w_i^{k_2} \ldots w_i^{k_p} F_{ir_1} F_{ir_2} \ldots F_{ir_q} = O(1)$$

Gaussian Equivalence Theorem (GET)

$$u_i = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} C_r F_{ri}$$

$$X_i = f[u_i]$$

Inputs of hidden units:

$$\lambda^k = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^k f[u_i]$$

GET in a nutshell: in the thermodynamic limit (with extensive latent dimension of the hidden manifold, $R = \gamma N$), the inputs of hidden units have Gaussian distribution. Then the model is solvable.

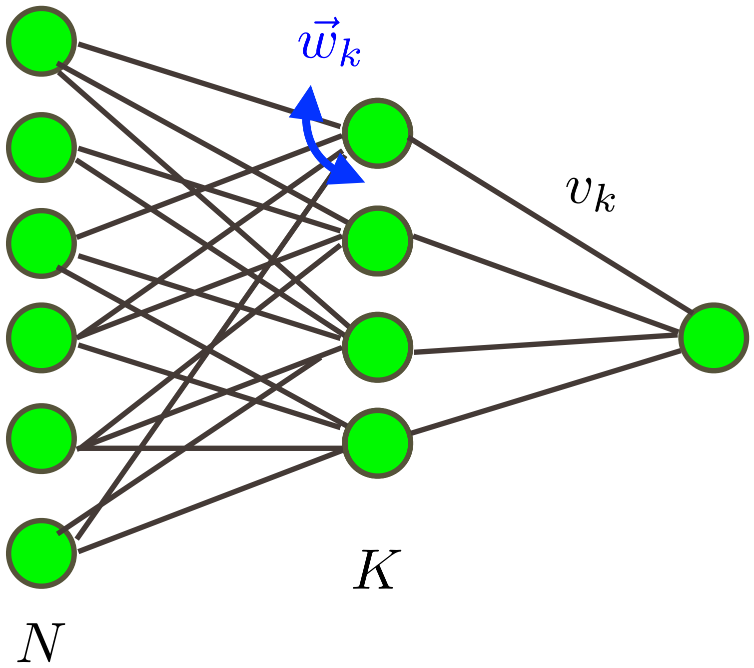
NB: depends on the manifold folding function f only through the three quantities

$c = \langle f(u)^2 \rangle$ $\quad a = \langle f(u) \rangle$ $\quad b = \langle u f(u) \rangle$ $\quad u$ Gaussian $\mathcal{N}(0,1)$

Any folding function f is statistically equivalent to a quadratic one

$$f(u) = \alpha + \beta u + \gamma u^2$$

Online learning of Hidden Manifold Model



Learn using a 2-layer neural net, K hidden units

$$\Phi\left(\vec{X}\right) = \sum_{k=1}^{K} g\left(\vec{w}^k . \vec{X}/\sqrt{N}\right)$$

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

$\vec{X}$ = inside hidden R-dimensional manifold, folded by function f

Desired output given constructed from latent representation

$$\Phi_t(\vec{X}) = \sum_{m=1}^{M} \tilde{g}\left(\sum_{r=1}^{R} \tilde{w}_r^m C_r\right)$$

Online learning: ODE for SGD

Evolution of the weights during learning

D Saad and S Solla 95, Biehl and Schwarze 95, …

$$\left(w_i^k\right)^{\mu+1} - \left(w_i^k\right)^{\mu} = -\frac{\eta}{\sqrt{N}} \Delta g'(\lambda^k) f(u_i)$$

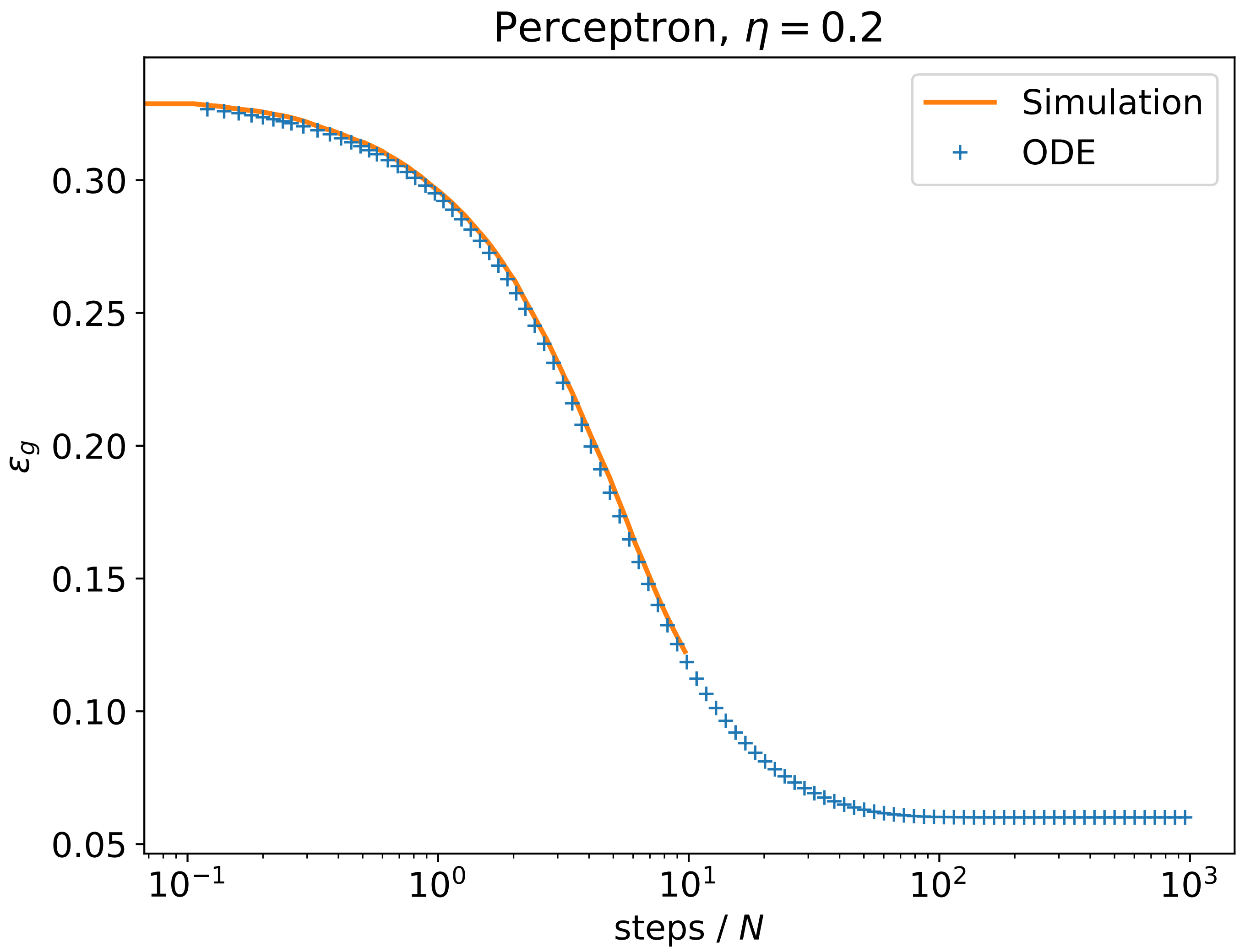
$$\Delta = \sum_{\ell=1}^{K} g(\lambda^\ell) - \sum_{m=1}^{N} \tilde{g}(\nu^m)$$

New pattern (and therefore new latent representation C_r) at each time

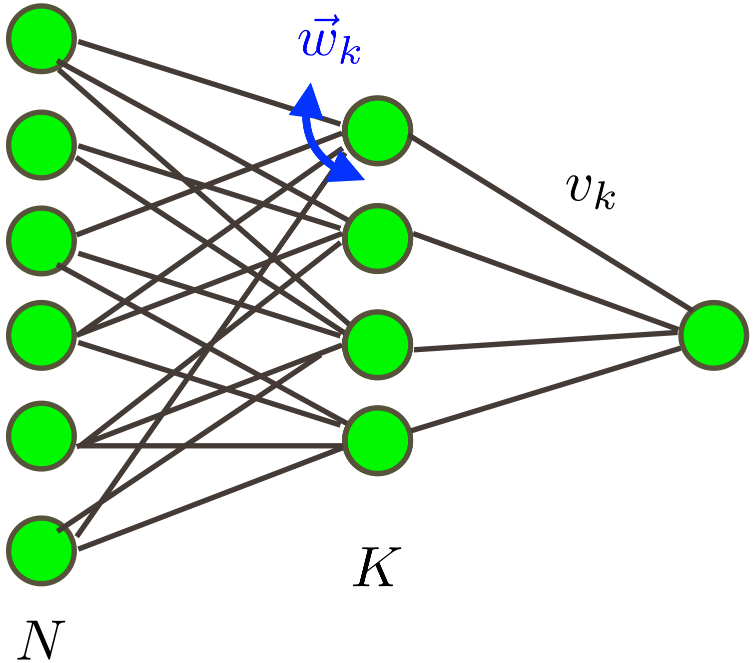
GET: λ^k and ν^m are Gaussian, and the learning dynamics can be analyzed by ordinary differential equations for order parameters like

$$W^{k\ell} \equiv \frac{1}{N} \sum_{i=1}^{N} w_i^k w_i^\ell$$

Preliminary result



Phase diagram of Hidden Manifold Model



Learn using a 2-layer neural net, K hidden units

$$\Phi\left(\vec{X}\right) = \sum_{k=1}^{K} g\left(\vec{w}^k.\vec{X}/\sqrt{N}\right)$$

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

$\vec{X}$ = inside hidden R-dimensional manifold, folded by function f

Desired output given constructed from latent representation

$$\Phi_t(\vec{X}) = \sum_{m=1}^{M} \tilde{g}\left(\sum_{r=1}^{R} \tilde{w}_r^m C_r\right)$$

Learn from database of P patterns.

Training error

$$E = \sum_{\mu=1}^{P} \epsilon\left[\Phi_t(X_\mu) - \Phi(X_\mu)\right]$$

Gardner's computation: probability (or volume) that w_i^k compatible with the data $\left\{ \vec{X}_\mu, \Phi_t(\vec{x}_\mu) \right\}$

$$Z = \int \prod_{i,k} \left[dw_i^k P_w(w_i^k) \right] e^{-\beta \sum_\mu \epsilon(\Phi_t(\vec{X}_\mu) - \Phi(\vec{X}_\mu))}$$

Gardner's computation: probability (or volume) that w_i^k compatible with the data $\left\{ \vec{X}_\mu, \Phi_t(\vec{x}_\mu) \right\}$

$$Z = \int \prod_{i,k} \left[dw_i^k P_w(w_i^k) \right] e^{-\beta \sum_\mu \epsilon(\Phi_t(\vec{X}_\mu) - \Phi(\vec{X}_\mu))}$$

Compute $\frac{1}{N} \log Z$ averaged over the distribution of

latent components $C_{\mu r}$, using replicas $\mathbb{E}_C Z^n \simeq e^{N\Psi(n)}$

$$\mathbb{E}_C \frac{1}{N} \log Z = \Psi'(0)$$

Gardner's computation: probability (or volume) that w_i^k compatible with the data $\left\{\vec{X}_\mu, \Phi_t(\vec{x}_\mu)\right\}$

$$Z = \int \prod_{i,k} \left[dw_i^k P_w(w_i^k)\right] e^{-\beta \sum_\mu \epsilon(\Phi_t(\vec{X}_\mu) - \Phi(\vec{X}_\mu))}$$

Compute $\frac{1}{N}\log Z$ averaged over the distribution of

latent components $C_{\mu r}$, using replicas $\mathbb{E}_C Z^n \simeq e^{N\Psi(n)}$

$$\mathbb{E}_C \frac{1}{N}\log Z = \Psi'(0)$$

$$Z^n = \int \prod_{ik} \prod_{a=1}^{n} \left[dw_i^{ka} P_w(w_i^{ka})\right] e^{-\beta \sum_{\mu,a} \epsilon(\Phi_t(\vec{X}_\mu) - \Phi^a(\vec{X}_\mu))}$$

Committee with weights w_i^{ka}

$$Z^n = \int \prod_{ik} \prod_{a=1}^{n} \left[dw_i^{ka} P_w(w_i^{ka}) \right] e^{-\beta \sum_{\mu,a} \epsilon(\Phi_t(\vec{X}_\mu) - \Phi^a(\vec{X}_\mu))}$$

$$\Phi^a\left(\vec{X}_\mu\right) = \sum_{k=1}^{K} g\left(\vec{w}^{ka}.\vec{X}_\mu / \sqrt{N}\right)$$

Natural variables = inputs to hidden neurons

$\lambda_\mu^{ka} = \frac{1}{\sqrt{N}} \sum_{i=1}^{N} w_i^{ka} f\left[\frac{1}{\sqrt{R}} \sum_{r=1}^{D} F_{ir} C_{\mu r}\right]$	$\nu_\mu^m = \frac{1}{\sqrt{R}} \sum_{r=1}^{R} \widetilde{w}_r^m C_{r\mu}$

GET ➡ These are joint Gaussian, with known covariance

$$\mathbb{E}_C Z^n = \int \prod_{ika} \left[dw_i^{ka} P_w(w_i^{ka}) \right] \prod_\mu \mathbb{E}_{\lambda;\nu} \exp\left[-\beta \sum_{\mu,a} \epsilon\left(\sum_m \widetilde{g}(\nu_\mu^m) - \sum_k g(\lambda_\mu^{ka})\right)\right]$$

➡ The replica computation can be done, for any $\epsilon, g, \tilde{g}\ , K, M$

In short

Gardner's computation: volume of space in w_i^k compatible with the data $\left\{\vec{X}_\mu, \Phi_t(\vec{x}_\mu)\right\}$

Evaluated with replicas

The volume can be written in terms of the local input fields to the hidden variables, λ_μ^{ka} .

The GET shows that these are Gaussian variables, independent for different patterns, correlated for one given pattern. Finite number of correlations between nk variables, so the computation can be done.

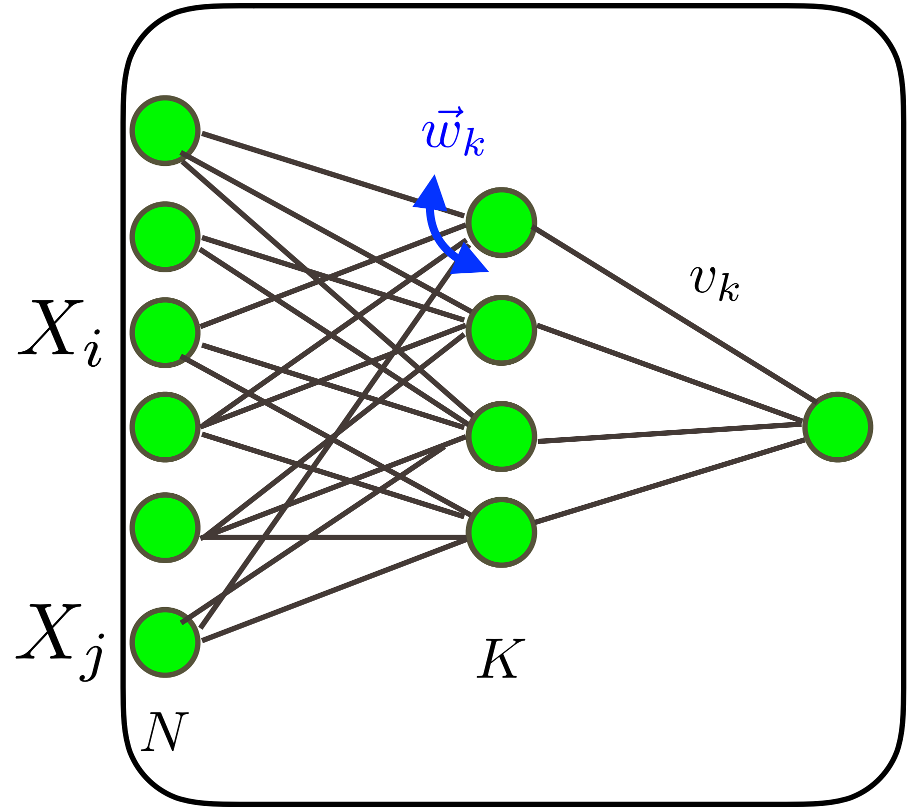
Results… coming soon.

NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid

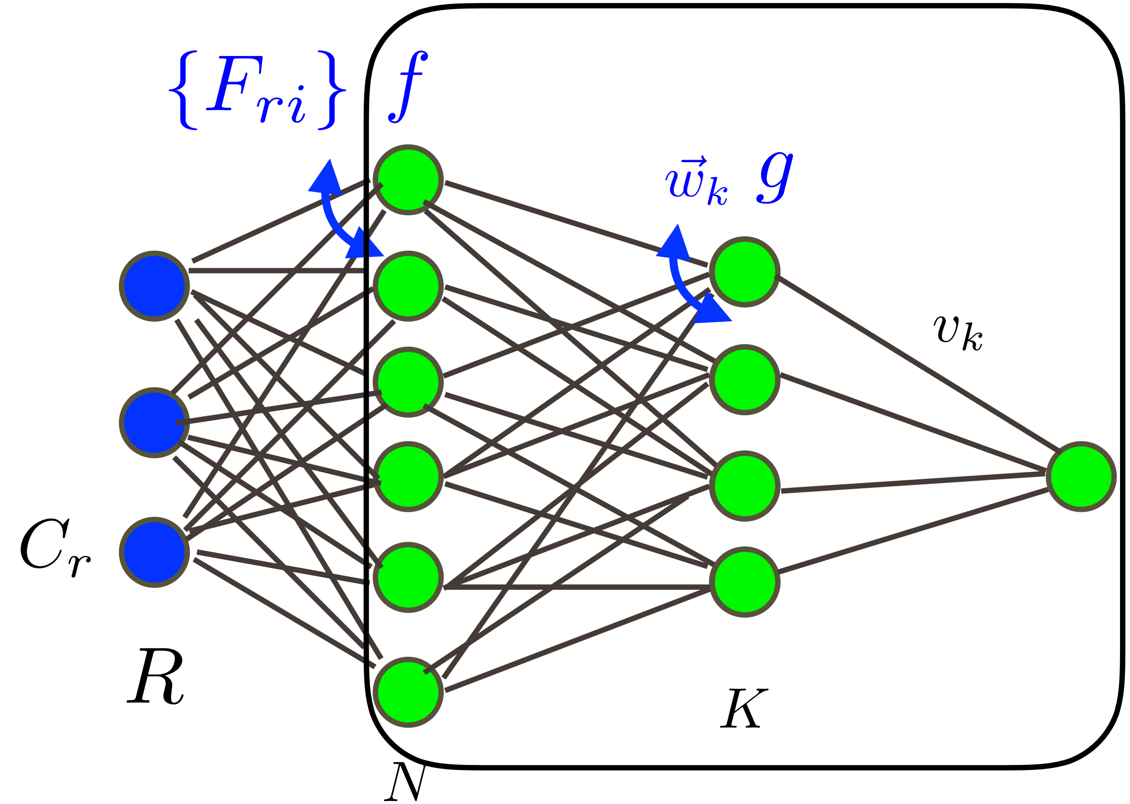


NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



Connection between C_r and X_i : F_{ri}

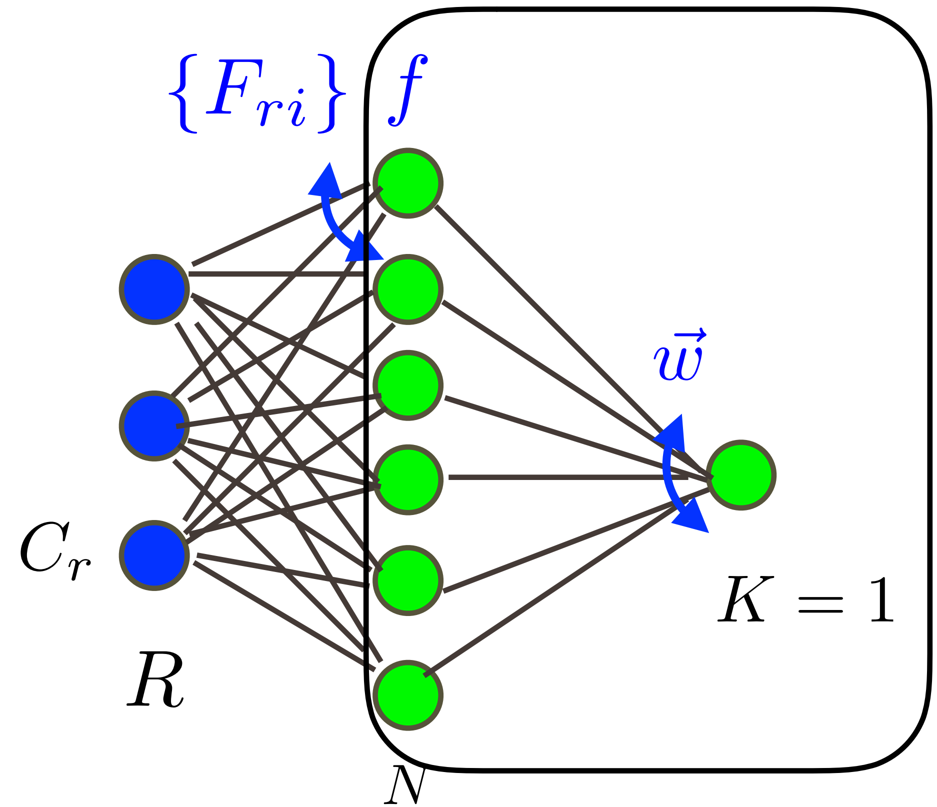
Hidden manifold model = build patterns directly in feature space, from iid coefficients in latent representation

NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



Connexion to Montanari Mei arXiv:1908.05335

Task $\Phi_t(\vec{X}) = \sum_{r=1}^{R} \tilde{w}_r^m C_r$

$$\Phi_t(\vec{X}) = \sum_{m=1}^{M} \tilde{g}\left(\sum_{r=1}^{R} \tilde{w}_r^m C_r\right)$$ with $M = 1$ linear $\tilde{g}$

Linear regression

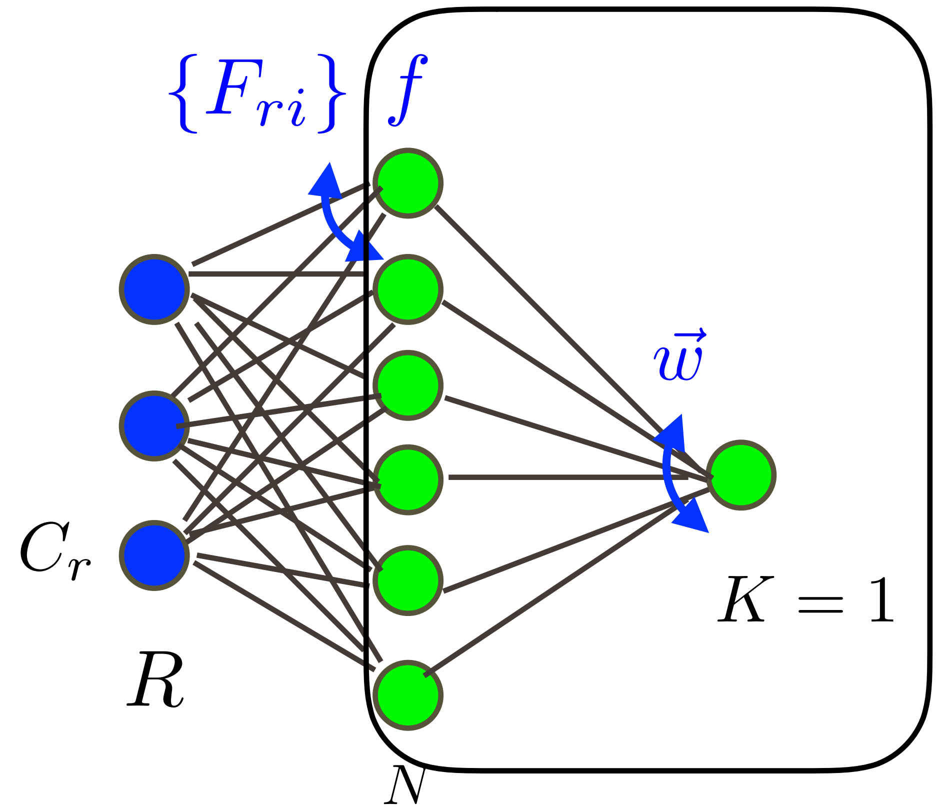
$$\Phi\left(\vec{X}\right) = \vec{w}^k . \vec{X}/\sqrt{N}$$

NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



Connexion to Montanari Mei
arXiv:1908.05335

Task $\Phi_t(\vec{X}) = \sum_{r=1}^{R} \tilde{w}_r^m C_r$

$$\Phi_t(\vec{X}) = \sum_{m=1}^{M} \tilde{g}\left(\sum_{r=1}^{R} \tilde{w}_r^m C_r\right)$$

with $M = 1$
linear $\tilde{g}$

Linear regression

$$\Phi\left(\vec{X}\right) = \vec{w}^k . \vec{X} / \sqrt{N}$$

Linear regression of random features is a special case of HMM

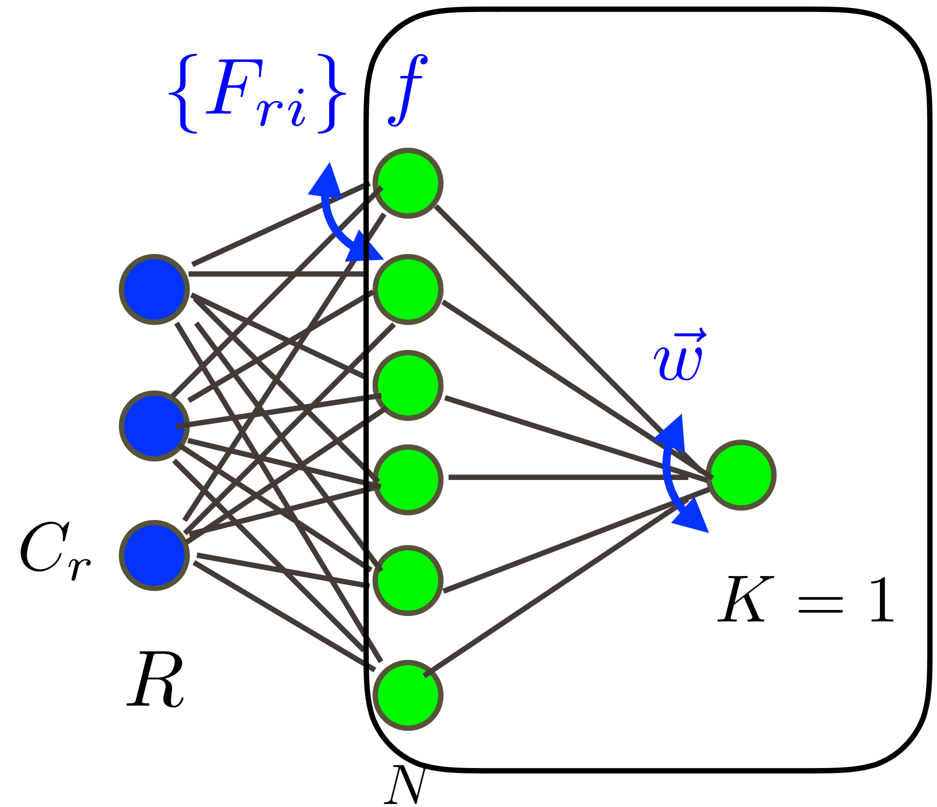
NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid

Connexion to Montanari Mei
arXiv:1908.05335



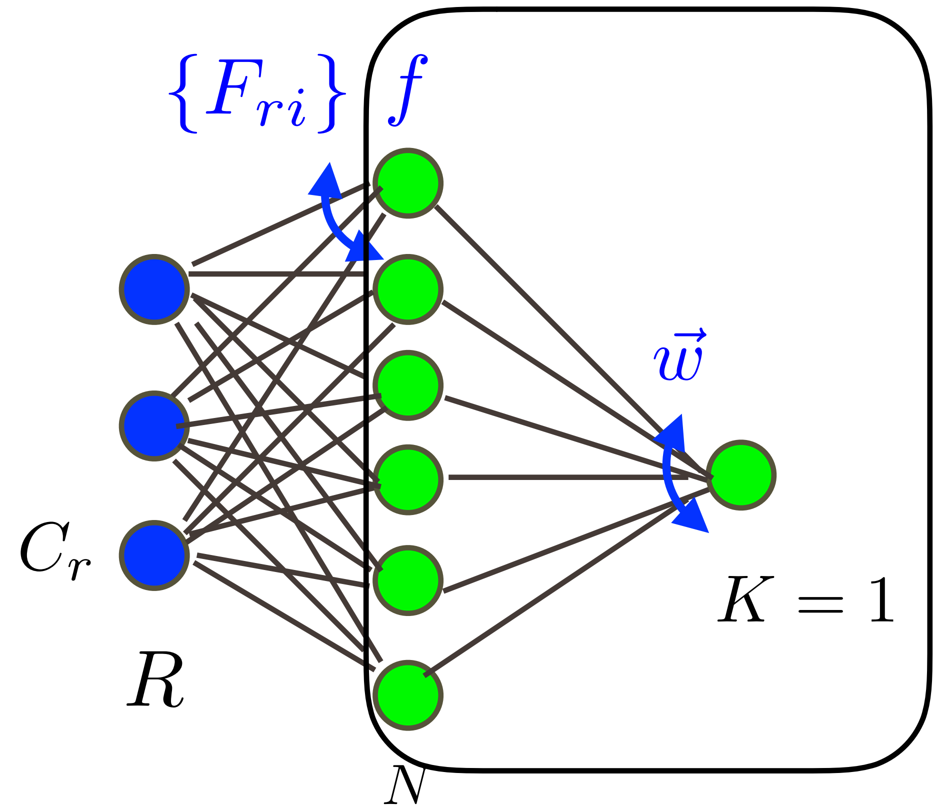
Linear regression

NB: Hidden manifold and random features

$$\vec{X} = f\left[\frac{1}{\sqrt{R}}\sum_{r=1}^{R} C_r \vec{F}_r\right]$$

Correlated components

iid



Linear regression

Connexion to Montanari Mei arXiv:1908.05335

Statistically equivalent to a case where

$$X_{\mu i} = \alpha + \frac{\beta}{\sqrt{R}}\sum_{r=1}^{R} C_{\mu r} F_{ri} + \eta_{\mu i}$$

iid

Consequence of GET and

$$c = \langle f(u)^2 \rangle \qquad a = \langle f(u) \rangle \quad b = \langle u f(u) \rangle$$

NB: applies also to the case where F_{ri} are not random (but they must be « balanced »)

Summary

Data structure is important

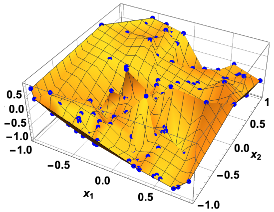
- Hidden manifolds and sub manifolds
- Combinatorial structure

Hidden Manifold Model

Data has « Latent representation »: $\{C_r\}$

Desired output (task) = function of latent representation

Example $y = g\left(\sum_{r=1}^{R} \tilde{w}_r C_r\right)$ $\vec{X} = f\left[\frac{1}{\sqrt{R}} \sum_{r=1}^{R} C_r \vec{F}_r\right]$



- Does not have the pathologies of teacher-student setup with iid data
- Learning and generalization phenomenology $\sim$ MNIST
- Can be studied analytically : online learning and full batch in the limit where $R = O(N)$, thanks to a Gaussian Equivalence property