A Principle of Least Action for the Training of Neural Networks

Skander Karkar***, Ibrahim Ayed***, Emmanuel de Bézenac*, Patrick Gallinari

Criteo AI Lab & Thales Theresis Lab & Sorbonne Université, Paris, France

Tau group seminar 2020

Introduction

- Neural networks are highly overparametrized.
- They systematically achieve nearly 0 training loss, yet generalise well to unseen data [4],
- This suggests the complexity of the network automatically adapts to the data
- This adaptivity is not captured by classical generalization bounds [1, 2].

Introduction

- Implicit biases present in the architecture, initialization and optimization algorithm are essential to its good generalization.
- A framework for explaining this generalisation should be able to take into account these biases

Contributions

- Through the dynamical viewpoint, we highlight the *low-energy bias* of residual networks.
- We formulate a Least Action Principle for the training of Neural Networks.
- We prove existence and regularity results for networks with minimal energy.
- We provide an algorithm for retrieving minimal energy networks compatible with different architectures.
- We show on standard classification tasks that our approach leads to a better generalization performance, without complexifying the architecture and especially in low data regimes.

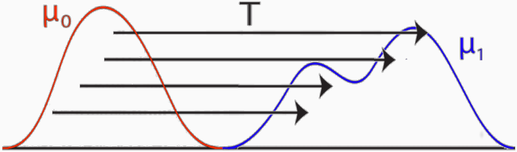
Table of contents

Optimal transport

Optimal transport: Monge formulation

- The problem of moving mass from one configuration to another with minimal total effort:

$$\begin{aligned} \inf_{T:X\to Y} \quad & \int_X c(x, T(x))\mathrm{d}\mu(x) \\ \text{subject to} \quad & T_{\#}\mu = \nu \end{aligned} \tag{1}$$



- We will consider ground costs $c(x, y) = \|x - y\|^p$ with $p > 1$.

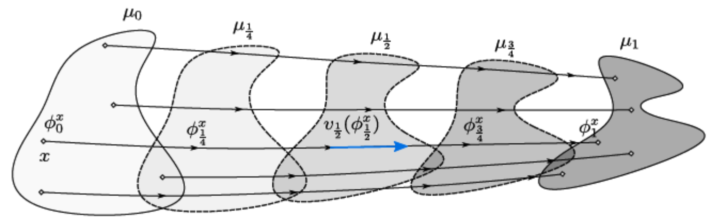
Optimal transport: dynamical formulations

- Equivalently, the density obeys the continuity equation in time:

$$\begin{aligned} \inf_{v} \quad & \int_0^1 \|v_t\|^p_{L^p(\mu_t)} \mathrm{d}t \\ \text{subject to} \quad & \partial_t \mu_t + \nabla \cdot (\mu_t v_t) = 0,\ \mu_0 = \mu,\ \mu_1 = \nu \end{aligned} \tag{2}$$

- Or the points move along a velocity field:

$$\begin{aligned} \inf_{v} \quad & \int_0^1 \|v_t\|^p_{L^p((\phi_t)_{\#}\mu)} \mathrm{d}t \\ \text{subject to} \quad & \partial_t \phi_t(x) = v_t(\phi_t(x)),\ \phi_0 = \mathrm{id},\ (\phi_1)_{\#}\mu = \nu \end{aligned} \tag{3}$$



General setting

Decomposing a neural network

- A neural network $f = F \circ T \circ \varphi$ is decomposed into 3 stages:
  1. **Dimensionality change:** φ transforms the input distribution $\mathcal{D}$ over $\mathbb{R}^n$ into distribution $\alpha = \varphi_\sharp \mathcal{D}$ over $\mathbb{R}^d$.
  2. **Data Transport:** α is transformed through a mapping $T : \mathbb{R}^d \to \mathbb{R}^d$, which we see as a transport map.
  3. **Task-specific final layers:** $F : \mathbb{R}^d \to \mathcal{Y}$ is applied to $T_\sharp \alpha$ in order to compute the loss $\mathcal{L}$ associated with the task at hand.
- Functions φ and F are often simple.
- If stages 1 and 2 are repeated many times, then many modern networks such as Wide ResNets and ResNexts fit this description.
- [3] finds that models that preserve dimension remain competitive.

Decomposing a neural network

- This leads us to define a *set of admissible targets* for the task:

$$S_{F,\mathcal{L}} = \{\beta \in \mathcal{P}(\mathbb{R}^d) \mid \mathcal{L}(F,\beta) = 0\} \tag{4}$$

- The goal of the learning task can then be reformulated as:

$$\text{Find } (T, F) \text{ such that } T_\sharp \alpha \in S_{F,\mathcal{L}} \tag{5}$$

Empirical analysis of transport dynamics in residual networks

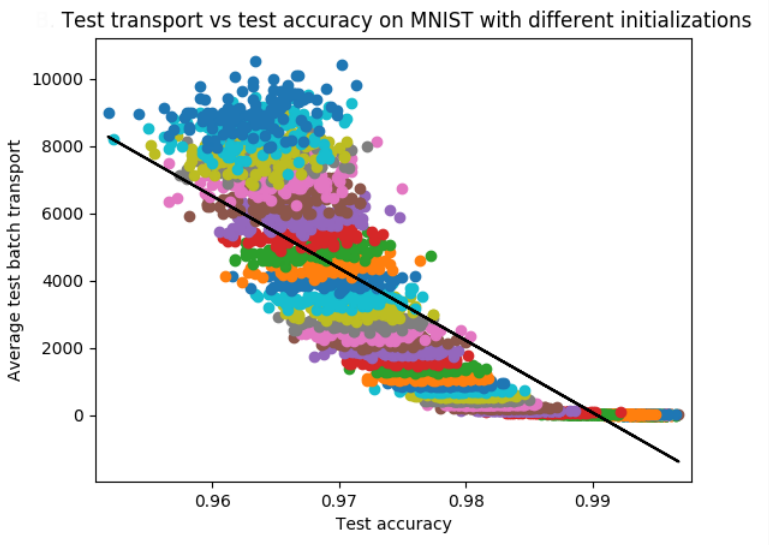
Residual networks

- A ResNet can be seen as a forward Euler scheme discretization of an associated ordinary differential equation

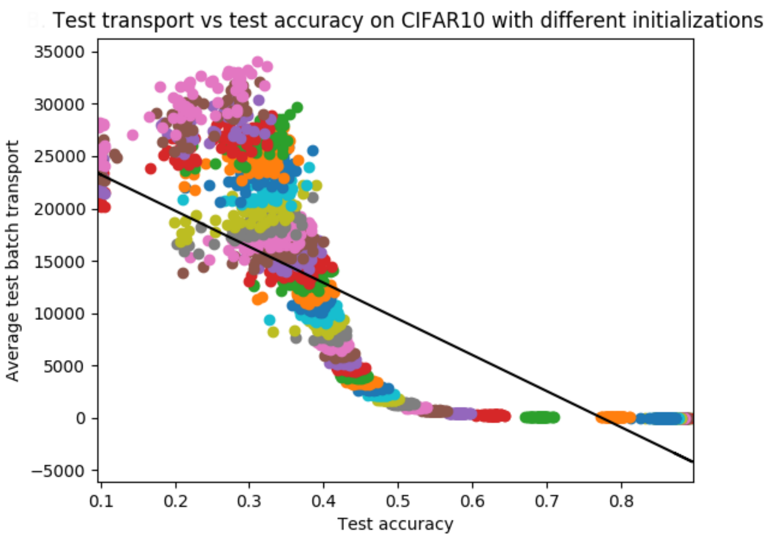
$$x_{k+1} = x_k + v_k(x_k) \longleftrightarrow \partial_t x_t = v_t(x_t)$$

- This allows to link ResNets to the optimal transport problem via optimal transport's dynamical formulation as a differential equation.
- If the data transport T is made up of residual blocks, then the transport cost is $\mathcal{C} = \sum_k \|v_k(x_k)\|^p$

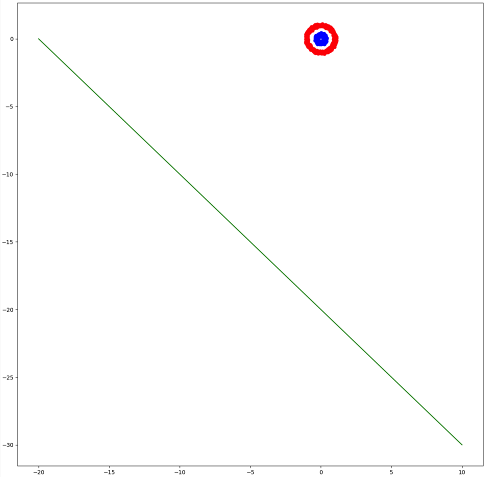
Empirical observations on MNIST



Empirical observations on CIFAR10



Empirical observations in 2 dimensions



Empirical observations in 2 dimensions

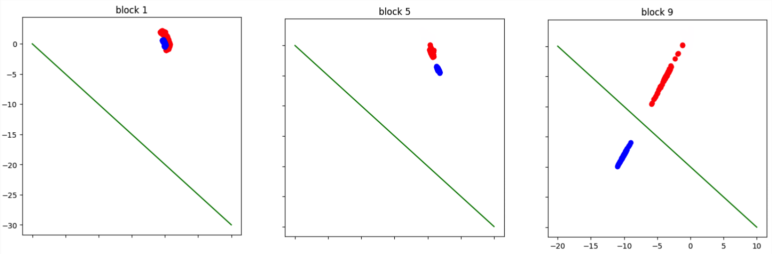


Figure 1: Transformed circles test set after each block after training

Empirical observations in 2 dimensions

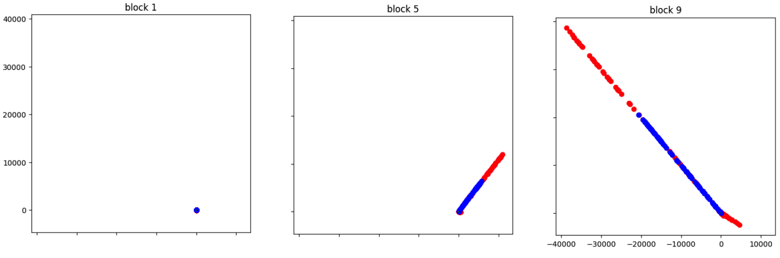


Figure 2: Transformed circles test set after each block after training with $\mathcal{N}(0,5)$ initialization

Empirical observations in 2 dimensions

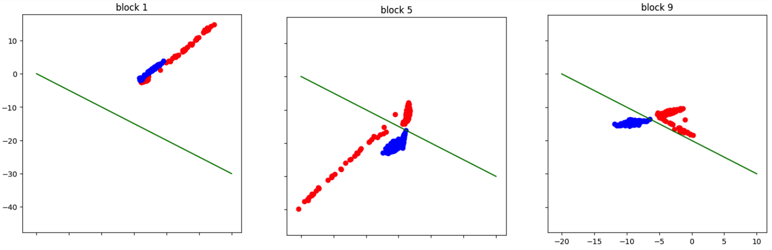


Figure 3: Transformed circles test set after each block after training with $\mathcal{N}(0, 5)$ initialization and batch normalization

Empirical observations in 2 dimensions

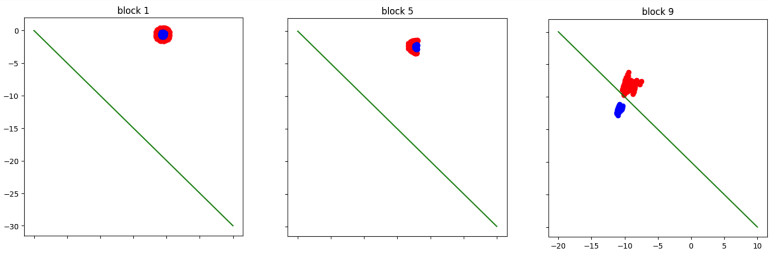


Figure 4: Transformed circles test set after each block after training with $\mathcal{N}(0,5)$ initialization, batch normalization and transport regularization

Least action principle for training neural networks

Least action networks: Formulation

- The empirical observations suggest trying to solve

$$\begin{aligned} \inf_{T,F} \quad & \mathcal{C}(T) = \int_{\mathbb{R}^d} c(x, T(x)) \mathrm{d}\alpha(x) \\ \text{subject to} \quad & T_\sharp \alpha \in S_{F,\mathcal{L}} \end{aligned} \tag{6}$$

- The equivalent dynamical formulation for $c(x, y) = \|x - y\|^p$ is

$$\begin{aligned} \inf_{v,F} \quad & \int_0^1 \|v_t\|^p_{L^p((\phi_t^\cdot)_\sharp \alpha)} \, \mathrm{d}t \\ \text{subject to} \quad & \partial_t \phi_t(x) = v_t(\phi_t(x)),\ \phi_0 = \mathrm{id},\ (\phi_1)_\sharp \alpha \in S_{F,\mathcal{L}} \end{aligned} \tag{7}$$

Least action networks: Theoretical results

Existence

- Under compacity assumptions, minimal energy mappings exist for both (6) and (7).
- In particular, for a minimizing $(T^\star, F^\star)$, $T^\star$ is an OT map between α and $T^\star_\sharp\alpha$.
- Uniqueness does not hold in general (as this is not a standard OT problem).

Least action networks: Theoretical results

Using (relatively) recent regularity results of OT mappings, minimal energy mappings inherit some regularity.

Let X, resp. Y, an open neighbourhood of the support of α, resp. $T^\star_\sharp \alpha$.

Regularity

- $T^\star$ is $\alpha-$ae differentiable.
- There exists A, resp. B, relatively closed in X, resp. Y, of null Lebesgue measure and $\eta > 0$ such that $T^\star \in C^{0,\eta}(X \setminus A, Y \setminus B)$.
- If α and $T^\star_\sharp \alpha$ are $C^{k,\eta}$ then $T^\star \in C^{k+1,\eta}(X \setminus A, Y \setminus B)$.

Least action networks: Discretization

- If we discretize the differential equation using an Euler scheme and the integrals using empirical measures we get

$$\begin{aligned} \min_{\theta} \quad & \mathcal{C}(\theta) = \sum_{x \in \mathcal{X}} \sum_{k=0}^{K-1} \|v_k(\phi_k^x)\|^p \\ \text{subject to} \quad & \phi_{k+1}^x = \phi_k^x + v_k(\phi_k^x), \phi_0^x = x, \mathcal{L}(\theta) = 0 \end{aligned} \tag{8}$$

  where $\mathcal{X}$ is the set of data points and θ parametrizes v_k and F.

- The first two conditions being trivially verified by a ResNet, the problem is equivalent to

$$\min_{\theta} \max_{\lambda > 0} \; \mathcal{C}(\theta) + \lambda \; \mathcal{L}(\theta) \tag{9}$$

Least action networks: Algorithm

- We use an algorithm inspired by the method of Multipliers:

$$\begin{cases} \theta_{i+1} = \arg\min_{\theta} \; \mathcal{C}(\theta) + \lambda_i \; \mathcal{L}(\theta) \\ \lambda_{i+1} = \lambda_i + \tau \; \mathcal{L}(\theta_{i+1}) \end{cases}$$

- The minimization is done via SGD for a predefined number of steps, starting from the previous parameter value θ_i.
- In practice, it is more stable to divide the objective by λ_i.

Experiments

Results on MNIST

Training set size	ResNet	LAP-ResNet
500	**90.8**, [90.4, 91.2]	**90.9**, [90.7, 91.1]
400	**88.4**, [88.0, 88.8]	**88.4**, [88.0, 88.8]
300	83.5, [83.0, 84.1]	**86.2**, [85.8, 86.6]
200	74.9, [73.9, 75.9]	**82.0**, [81.5, 82.5]
100	56.4, [54.9, 58.0]	**70.0**, [69.0, 71.0]

Table 1: Average highest test accuracy and 95% confidence interval of ResNet9 over 50 instances on MNIST with training sets of different sizes.

Results on CIFAR10

Training set size	ResNet	LAP-ResNet
50 000	91.49, [91.40, 91.59]	**91.94**, [91.84, 92.04]
30 000	88.61, [88.47, 88.75]	**89.41**, [89.31, 89.50]
20 000	85.73, [85.59, 85.87]	**86.74**, [86.61, 86.87]
10 000	79.25, [79.00, 79.49]	**80.90**, [80.74, 81.06]
5 000	70.32, [70.00, 70.63]	**72.58**, [72.36, 72.79]
4 000	67.80, [67.55, 68.07]	**70.12**, [69.81, 70.42]

Table 2: Average highest test accuracy and 95% confidence interval of ResNet9 over 20 instances on CIFAR10 with training sets of different sizes.

Results on CIFAR10

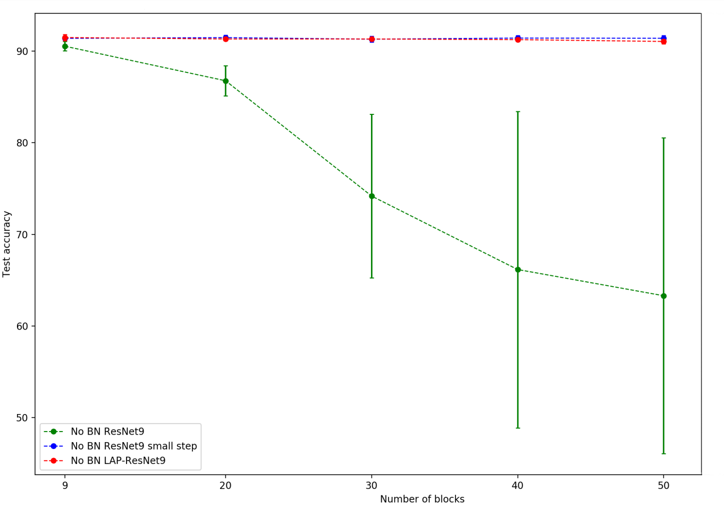


Figure 5: Test accuracy and 95% confidence interval of ResNet models of different depth without batch normalization on CIFAR10

Results on CIFAR100

Training set size	ResNeXt	LAP-ResNeXt
50 000	72.97, [71.79, 74.14]	**76.11**, [75.32, 76.89]
25 000	62.55, [60.18, 64.92]	**64.11**, [62.25, 65.96]
12 500	45.90, [43.16, 48.67]	**48.23**, [46.39, 50.07]

Table 3: Average highest test accuracy and 95% confidence interval of ResNeXt50 over 10 instances on CIFAR100 with training sets of different sizes.

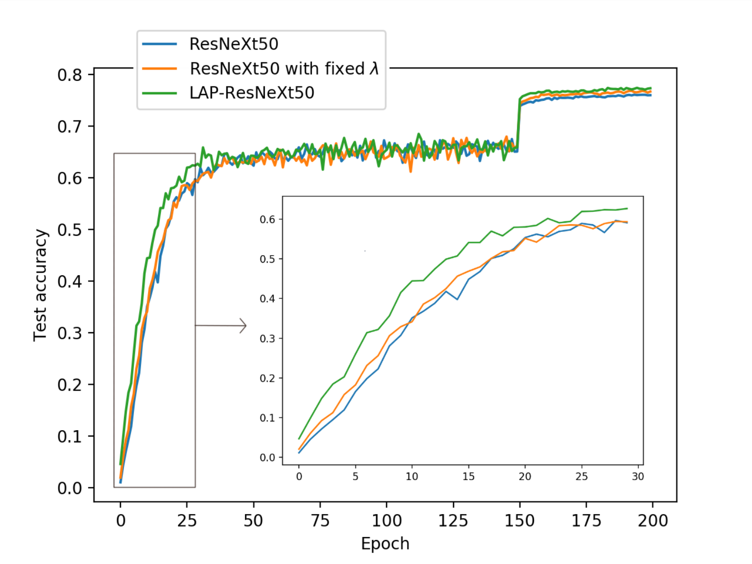


Figure 6: Test accuracy during training of ResNeXt50 models on CIFAR100.

Conclusion

Conclusion

- The least action principle improves test performance especially for small datasets, bad initializations and large networks that overfit.
- It does this without complexifying the architecture or slowing down the training.
- Linking this simple technique to optimal transport theory offers existence and regularity results.
- This regularity is confirmed in practice by increased stability, as seen in the narrower confidence intervals, but this remains to be explored further.

References

[1] Mikhail Belkin et al. **Reconciling modern machine-learning practice and the classical bias–variance trade-off.** *PNAS*, 2019.

[2] P. Nakkiran et al. **Deep double descent: Where bigger models and more data hurt.** In *ICLR*, 2020.

[3] M. Sandler et al. **Non-discriminative data or weak model? on the relative importance of data and model resolution.** In *ICCVW*, 2019.

[4] C. Zhang et al. **Understanding deep learning requires rethinking generalization.** In *ICLR*, 2017.