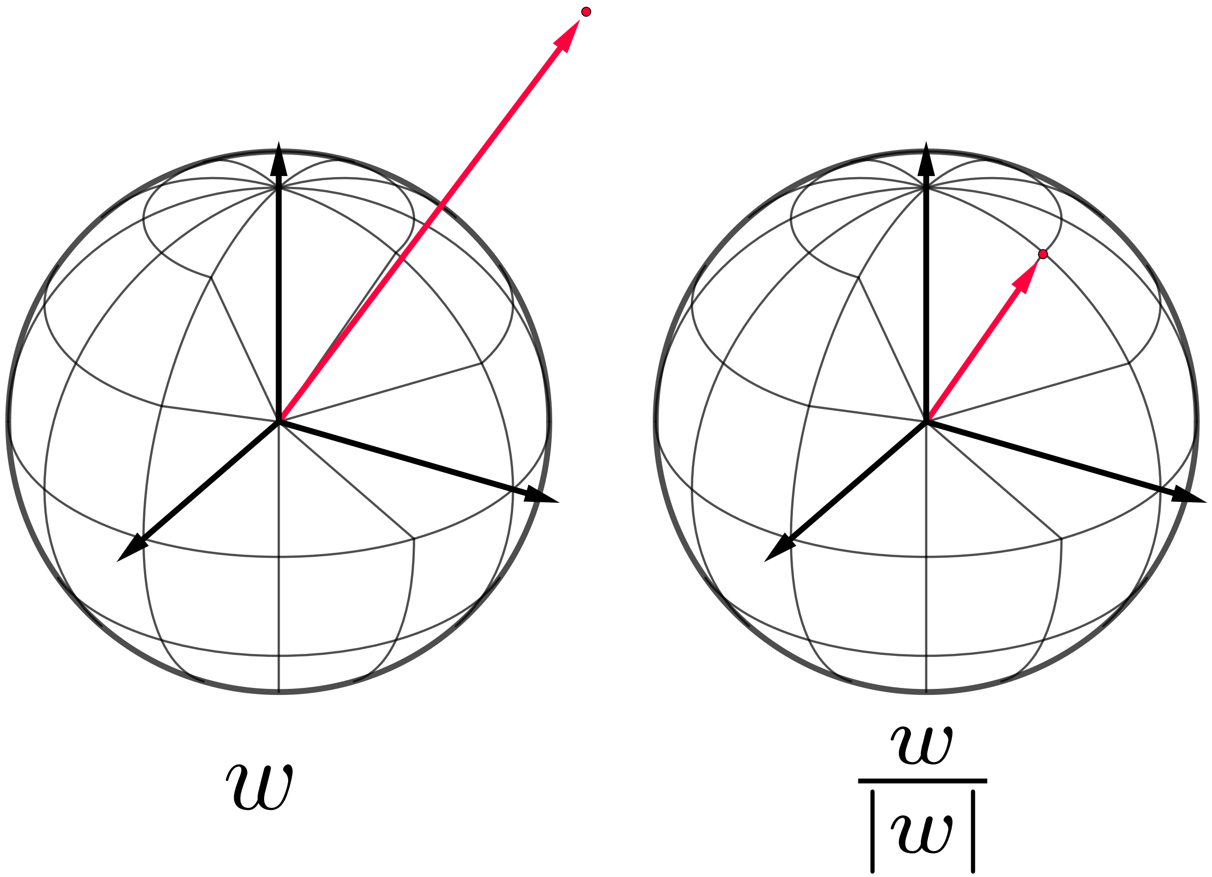
Deep Networks on Toroids: Removing Symmetries Reveals the Structure of Flat Regions in the Landscape Geometry

Fabrizio Pittorino, Antonio Ferraro, Gabriele Perugini,
***Christoph Feinauer**, Carlo Baldassi, Riccardo Zecchina*

Bocconi University, Milan, Italy

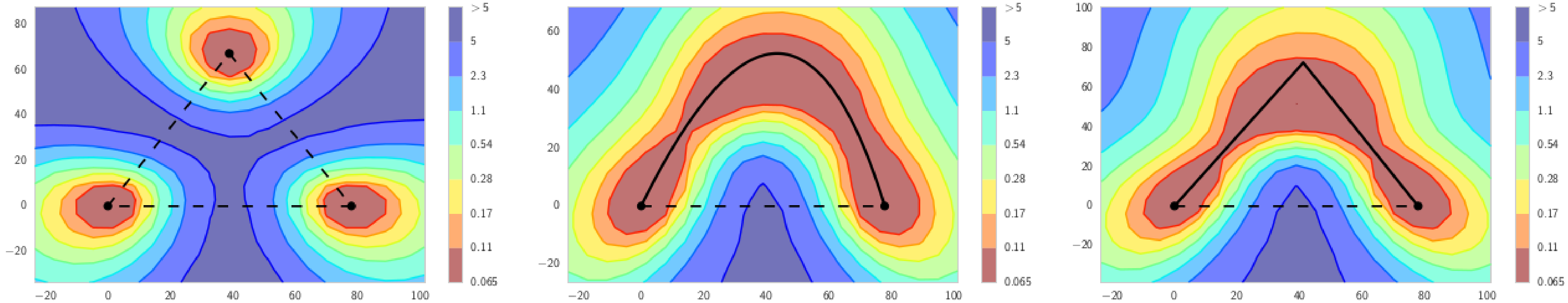
ICML 2022 - Short Presentation



Spherical coordinate diagram adapted from https://en.wikipedia.org/wiki/Spherical_coordinate_system

Neural Network Landscapes

- can be explored in terms of the *loss* or *error* space
- determine the dynamics of gradient descent
- exhibit non-trivial symmetries
- points at the same height can have drastically different generalization properties, connected to **flatness**
- minima found independently can often be connected with relatively simple paths



Plot from *Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., & Wilson, A. G. (2018). Loss surfaces, mode connectivity, and fast ensembling of dnns. Advances in neural information processing systems.*

Our work

- studies the error landscape
- ... after removing symmetries
- ... around solutions found with different algorithms (entropic algorithms and standard algorithms)
- ... in networks with continuous and binary weights

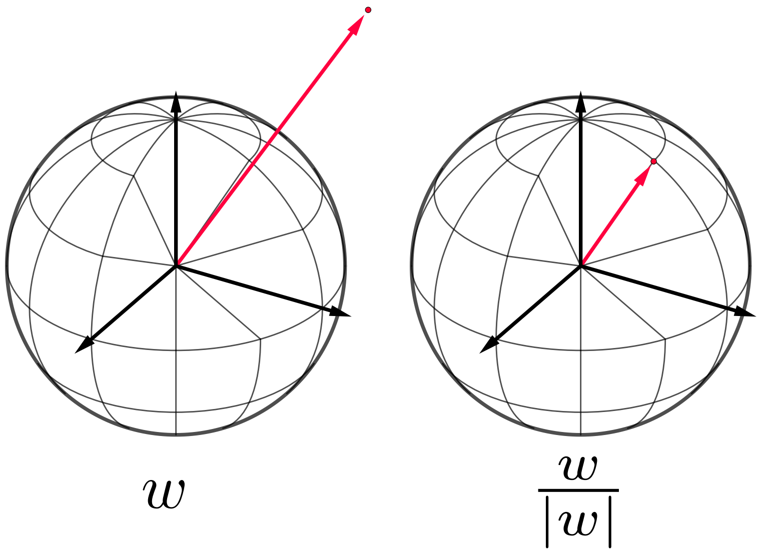
Symmetries

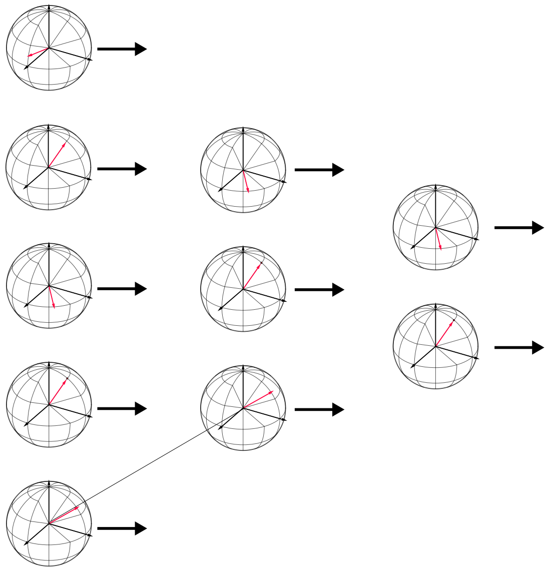
Weight-Norm

For the *ReLU* activation function we have $f(\alpha x) = \alpha f(x)$ for $\alpha \geq 0$ and therefore

$$f(\vec{w} \cdot \vec{x}) = |\vec{w}| f\left(\frac{\vec{w}}{|\vec{w}|} \vec{x}\right),$$

which means we can *push* the weight norms to the next layer.





Every neuron becomes a hypersphere with norm=1

The complete network is a product of hyperspheres

[The last layer precedes a `argmax` and since we are only interested in the error we can normalize it *globally* with a positive factor]

Symmetries

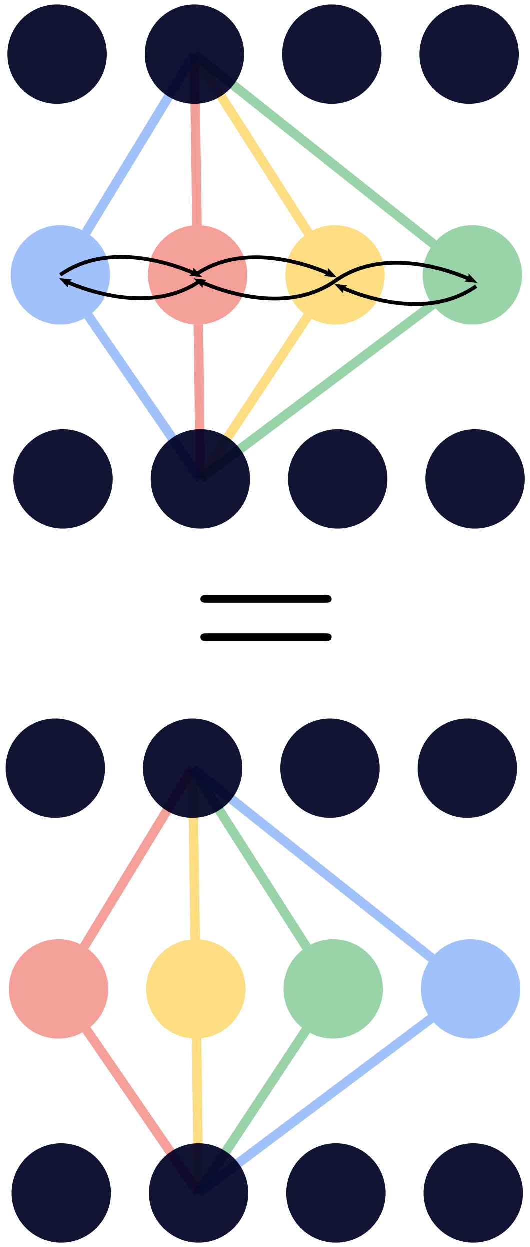
Permutation

- Neurons within a layer can be exchanged as long as incoming and outgoing connections are taken care of
- The same goes for kernels in convolutional layers
- Networks need to be *aligned* before comparing them

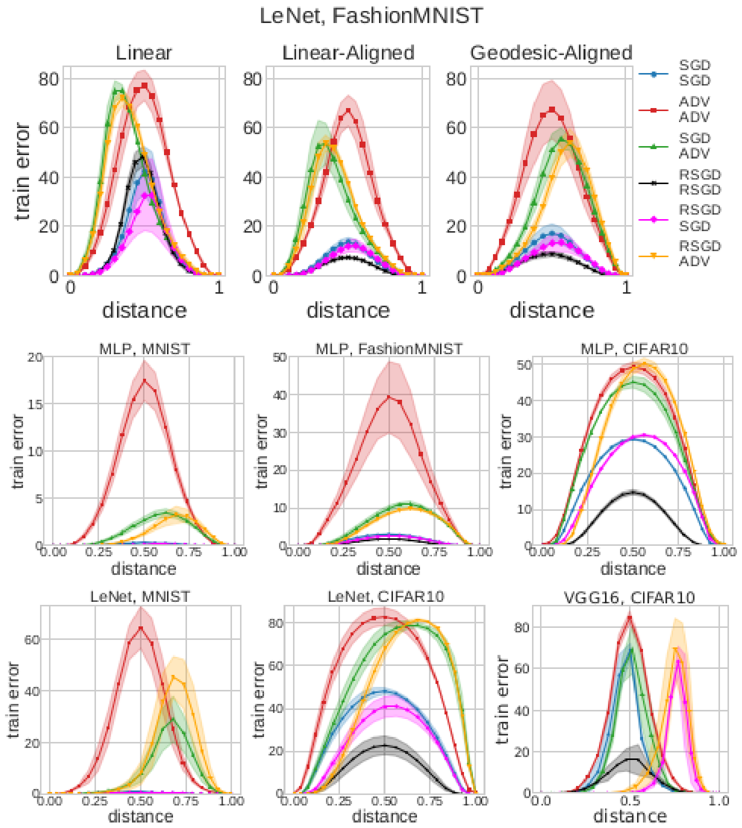
Simple Algorithm for Matching

```
Input: Two normalized NNs with parameters A[1..L], B[1..L] and L layers
for l = 1 to L - 1 do
    π = Match(A[l] , B[l])
    PermutePrev(B[l], π)
    PermuteNext(B[l+1] , π ` )
end for
```

`Match` takes two sets of parameter vectors and solves a weighted bipartpite graph matching problem using cosine similarities between vectors as weights.

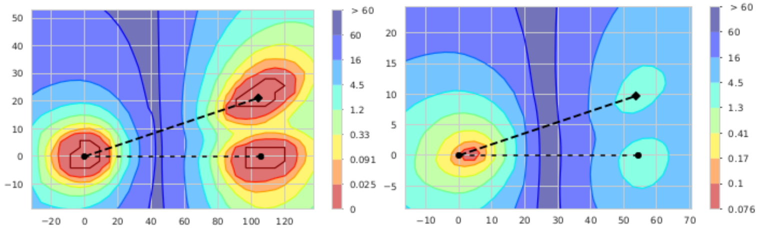


Mode Connectivity



- SGD: Stochastic Gradient Descent
- RSGD: *Replicated SGD* (finds flatter minima)
- ADV: Adversarial Initialization
- Linear: Straight path between minima
- Geodesic: Path on normalized sub-manifold

VGG16 on Cifar10



- Left Panel: Unnormalized
- Right Panel: Normalized
- Left Points: RSGD (finds flatter minima)
- Right Points: unaligned/aligned SGD with adversarial initialization

Difference is only visible *after* symmetry removal

Deep Networks on Toroids: Removing Symmetries Reveals the Structure of Flat Regions in the Landscape Geometry

Fabrizio Pittorino [1] **Antonio Ferraro** [1] **Gabriele Perugini** [1 2] **Christoph Feinauer** [1] **Carlo Baldassi** [1]
Riccardo Zecchina [1]