Separate or share? Hierarchical balancing of orthogonalization, alignment, and abstraction in continual learning

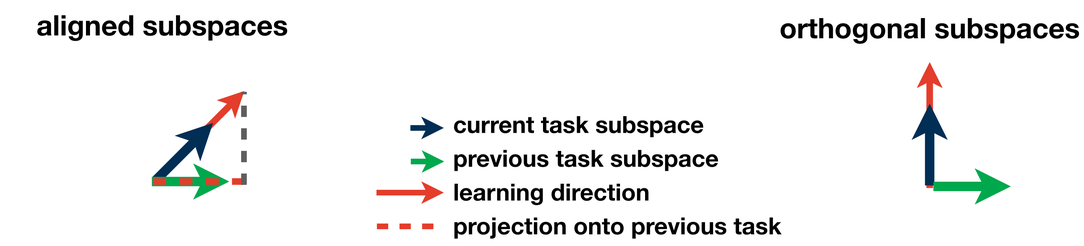
Andrea Albert[1], Gergő Orbán[1], Márton A. Hajnal[1†]

1) Department of Computational Sciences, MTA HUN-REN Wigner Research Centre for Physics, Budapest, Hungary
†) hajnal.marton@wigner.mta.hu

Motivation

Continual learning: acquiring tasks sequentially

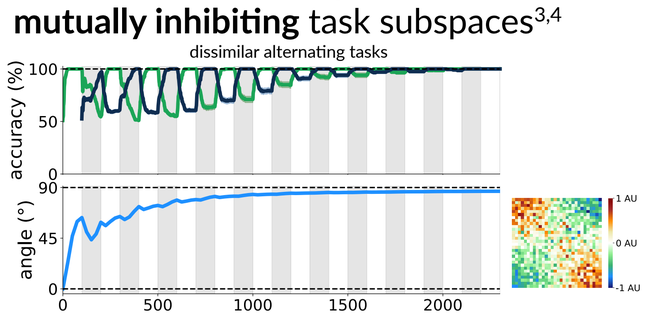
Avoid **catastrophic forgetting** with **orthogonal representations**[1,2]



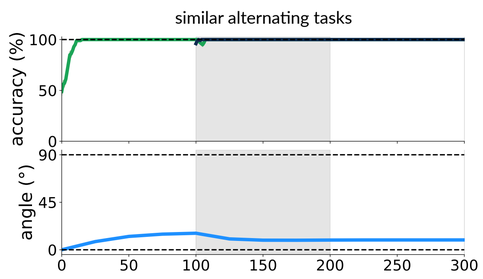
previous task projection > 0:
interfering weight updates

previous task projection ~ 0:
independent weight updates

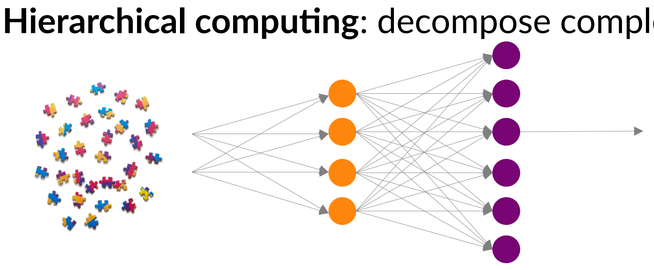
Representations of **dissimilar** tasks **spontaneously orthogonalize** with **mutually inhibiting** task subspaces[3,4]



Representations of **similar** tasks remain **aligned**[4]



Hierarchical computing: decompose complexity into simpler steps



Question:
How do **separate** and **shared** representations develop along the hierarchy?

Methods

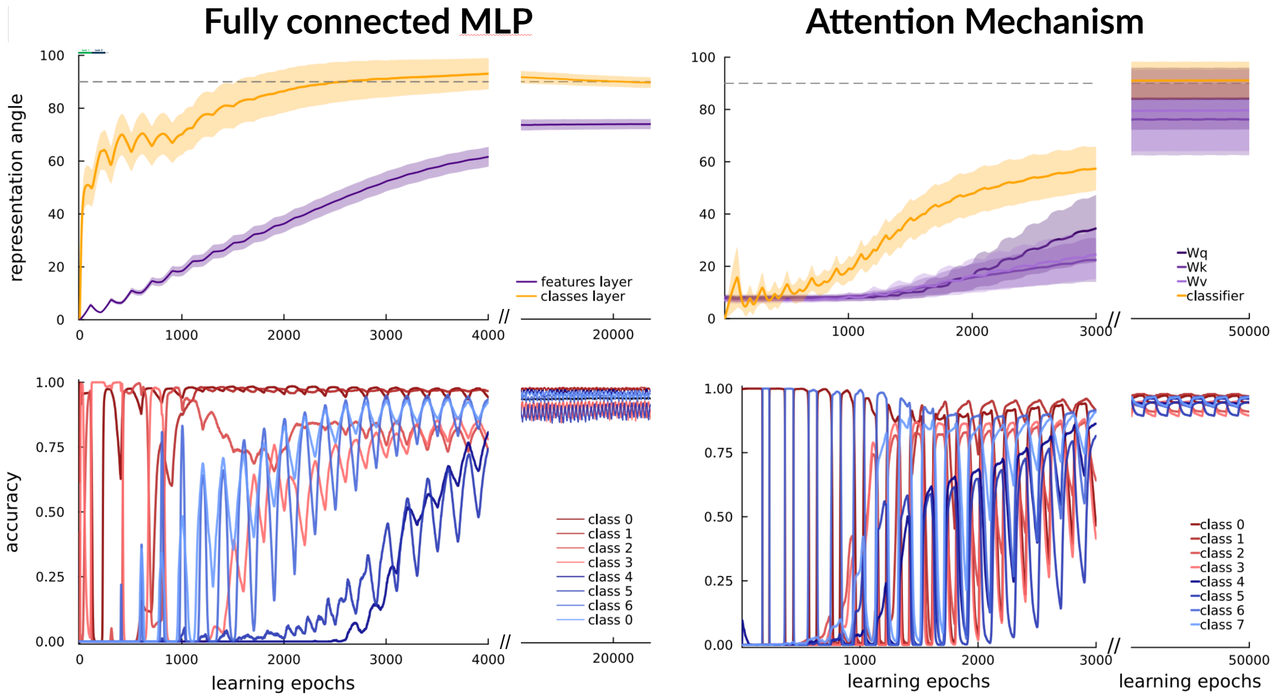
A **medium complexity task** (MNIST, handwritten digits classification) to understand **hierarchical representations** evolving from **aligned initial** subspaces.

Repeatedly alternating, packaged task **design**:
Task 1: classify 0,1,2,3 Task 2: classify 4,5,6,7 ⟳

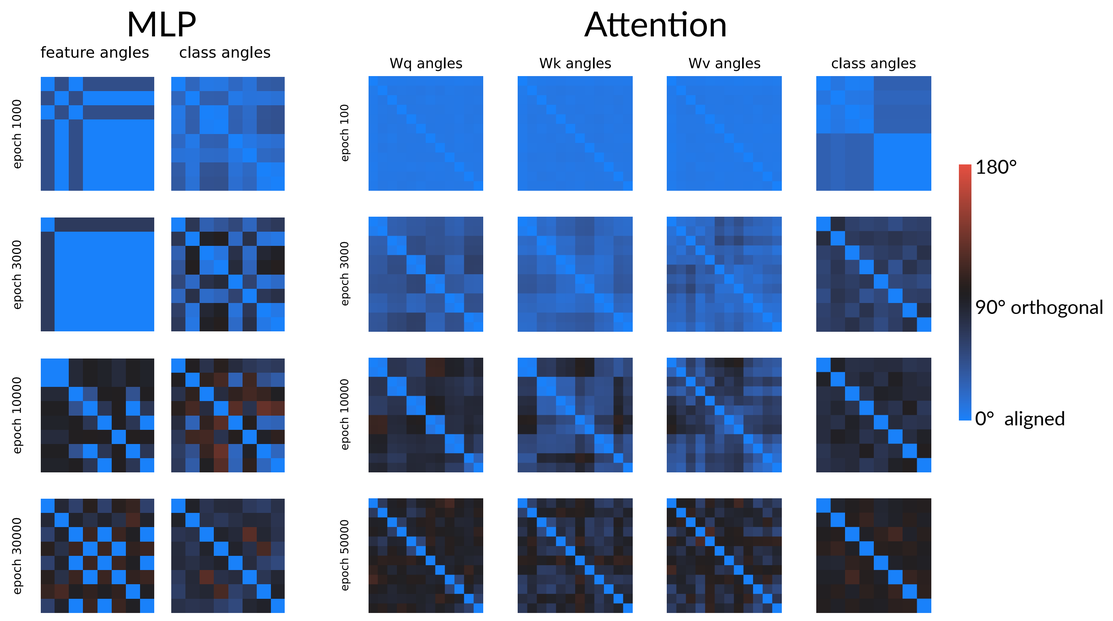
Two-layer **hierarchical classifier**, lower layer either **fully connected** or **attention**.

Results

1. **Representation** geometry **spontaneously reorganizes** to task alternation pressure: **hierarchical orthogonalization**
Early: tasks compete for **memory**; Late: tasks' memories do **not interfere**

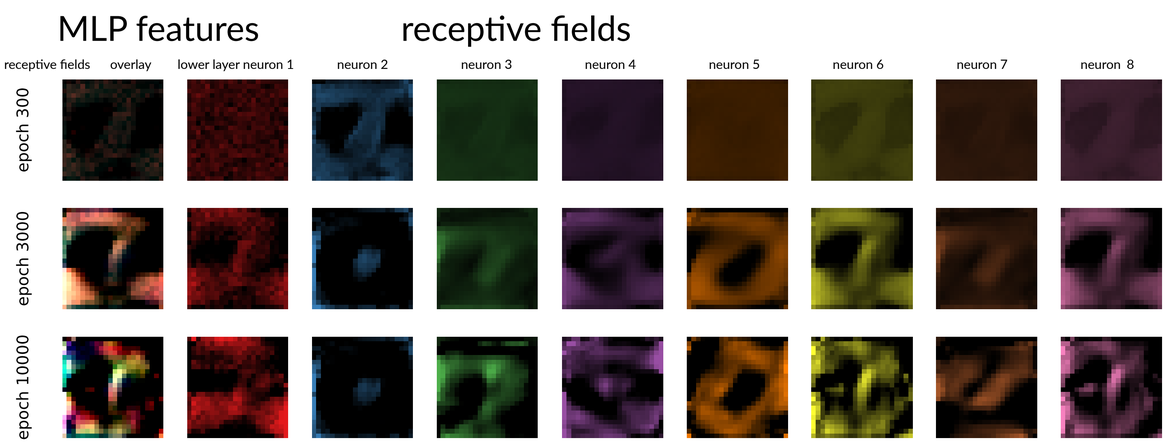


2. **Individual MLP** feature- and class-layer **units orthogonalize** **Attention** $W_{q,k,v}$ contextual projectors **orthogonalize,** channels across heads first, then within heads.

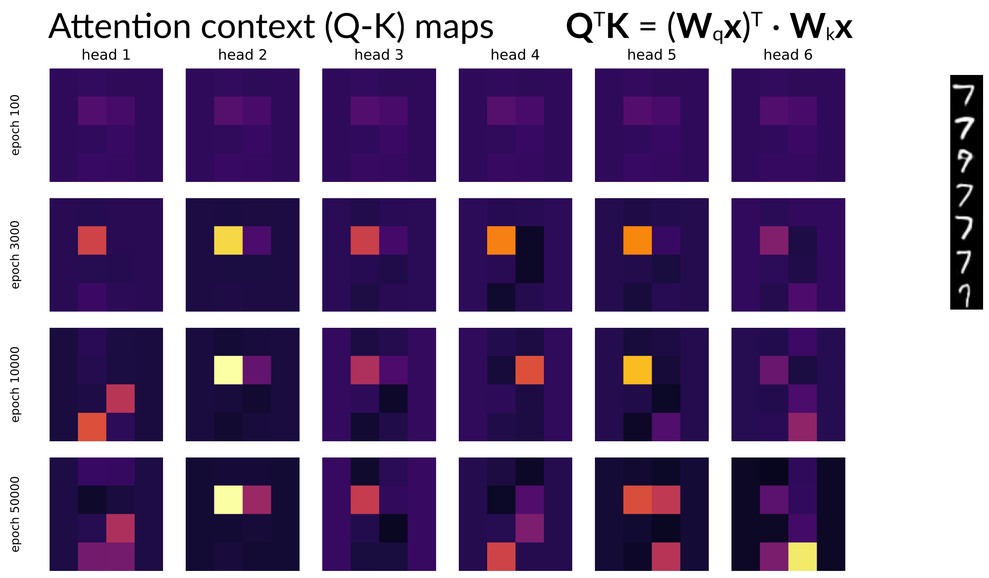


3. Classes **reuse features, redundancy disappears**

MLP. **Classes** first develop **private** features, then degenerate, redundant features **collapse** as the **feature layer fully orthogonalizes**. MLP features can be **tuned** to look "nice" (weak **size** + **smooth**ness regularizations)



Attention. Classes first use overlapping **patch interactions**, then progressively **separate** across heads, producing **orthogonal contextual computations**.



Summary

Continual learning **reorganizes hierarchical** networks **spontaneously**:

- Alternating tasks force **orthogonal task memories**
- **Orthogonal representations form** by **backward** computational pressure over the hierarchy, **higher** level representations **first**
- **Common** lower level **features** and **relations** are **shared across tasks**

Task **complexity** exponentially **increases** "wait" **time → compute hierarchically!**

Acknowledgement

Funding: Wigner RMI 2026
The authors thank Zsófia Pálffy for the inspiring discussions

References:
1) J. Kirkpatrick et al., *Overcoming catastrophic forgetting in neural networks*, PNAS, 114(13) 3521, 2017
2) L. Duncker, L. Driscoll, K. Shenoy, M. Sahani, and S. D, *Organizing recurrent network dynamics by task-computation to enable continual learning*, in NeurIPS, 2020
3) M.A. Hajnal et al., *Shifts in attention drive context-dependent subspace encoding in anterior cingulate cortex in mice during decision making*, Nature Communications, 15(1) 5559, 2024
4) A. Albert, G. Orbán, M.A. Hajnal, *Orthogonal task representations prevent catastrophic forgetting in continual learning*, spotlight, 6th Intl' Conference on Mathematics of Neuroscience and AI, 2025