Learning low-dimensional inputs for brain-machine interface control:

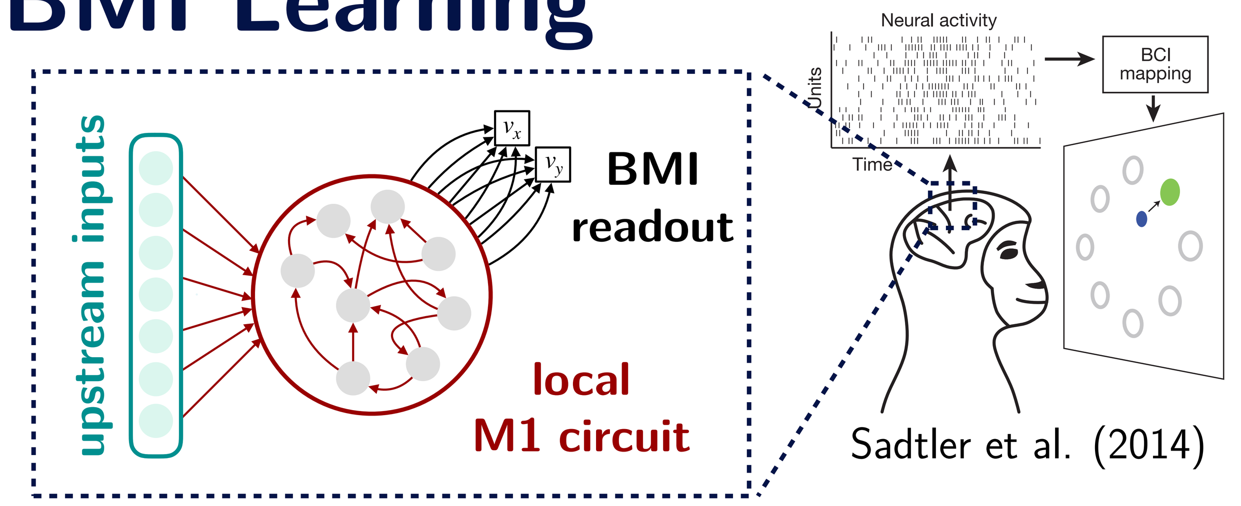
A motor cortical model of brain-machine interface learning, fast and slow

Jorge A. Menéndez[1,2] & Peter E. Latham[1]

[1]*Gatsby Computational Neuroscience Unit,* [2]*CoMPLEX; University College London*

UCL

BMI Learning



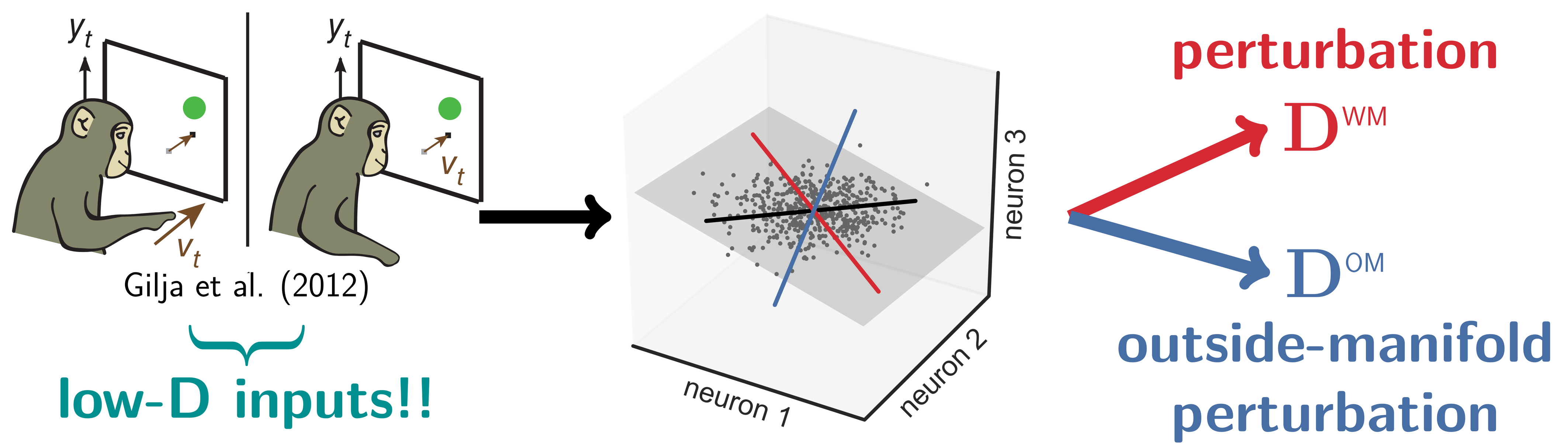
Observations:
- ☐ fast
- ☐ ~~gradients~~
- ☐ flexible

Re-wiring:
- × $\mathcal{O}(N^2)$ parameters
- × ⇒ slow
- × forgetful

Re-aiming:
- ✓ low-dimensional
- ✓ ⇒ fast
- ?

BMI learning, fast and slow

[Sadtler et al. (2014)]



SUMMARY

1) *Hypothesis*: BMI learning = **optimizing upstream inputs within a low-d manifold**

2) Can explain **behavioral** and **neural** observations of **fast-timescale BMI learning in M1** [Sadtler et al. '14, Golub et al. '18]

3) Such a mechanism could also underlie **slow-timescale BMI learning**, obviating the need to re-structure the local M1 circuit

Modelling re-aiming

Linear BMI decoder:

$$\mathbf{v}(t) = \mathbf{D}\mathbf{r}(t) + \mathbf{b}$$

Standard RNN dynamics:

$$\tau \dot{x}_i = -x_i + \sum_{j=1}^{N} W_{ij}^{\mathrm{rec}} r_j + \sum_{k=1}^{M} W_{ik}^{\mathrm{in}} u_k$$

$$r_i = \phi(x_i)$$

with low-dimensional inputs:

$$\mathbf{u} = f(\boldsymbol{\theta})$$

$\boldsymbol{\theta} \in \mathbb{R}^K$ parameterizes K-dimensional inputs

Approach: study optimal re-aiming solutions

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \underbrace{\|\mathbf{D}\mathbf{r}(t) - \mathbf{v}^*\|^2}_{E_t(\boldsymbol{\theta})} + \gamma \frac{\|\mathbf{u}\|^2}{M}$$

- * fix endpoint time t
- * assume static u_i

Linear case

$$\phi(x_i) = x_i, \quad f(\boldsymbol{\theta}) = \mathbf{M}\boldsymbol{\theta}$$

Solve linear differential equation:

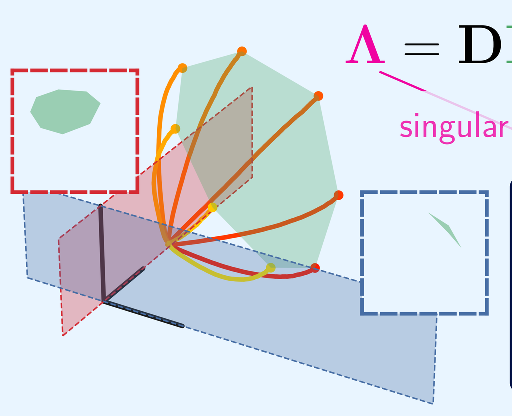
$$\begin{cases} \tau\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \\ \Rightarrow \mathbf{x}(t) = \mathbf{F}\mathbf{y} + \mathbf{F}_0\mathbf{x}(0) \\ \mathbf{F} = \left(e^{\mathbf{A}\frac{t}{\tau}} - \mathbf{I}\right)\mathbf{A}^{-1}\mathbf{B} \\ \mathbf{F}_0 = e^{\mathbf{A}\frac{t}{\tau}} \end{cases}$$

$$\mathbf{r}(t) = \mathbf{F}(t)\boldsymbol{\theta} + \mathbf{F}_0(t)\mathbf{r}(0)$$

⇒ re-aiming depends on endpoint alignment

$$\boldsymbol{\Lambda} = \mathbf{D}\mathbf{F}(t) = \begin{bmatrix} - & \mathbf{d}_1^T & - \\ - & \mathbf{d}_2^T & - \end{bmatrix} \begin{bmatrix} | & | & & | \\ \mathbf{f}_1 & \mathbf{f}_2 & \ldots & \mathbf{f}_K \\ | & | & & | \end{bmatrix}$$

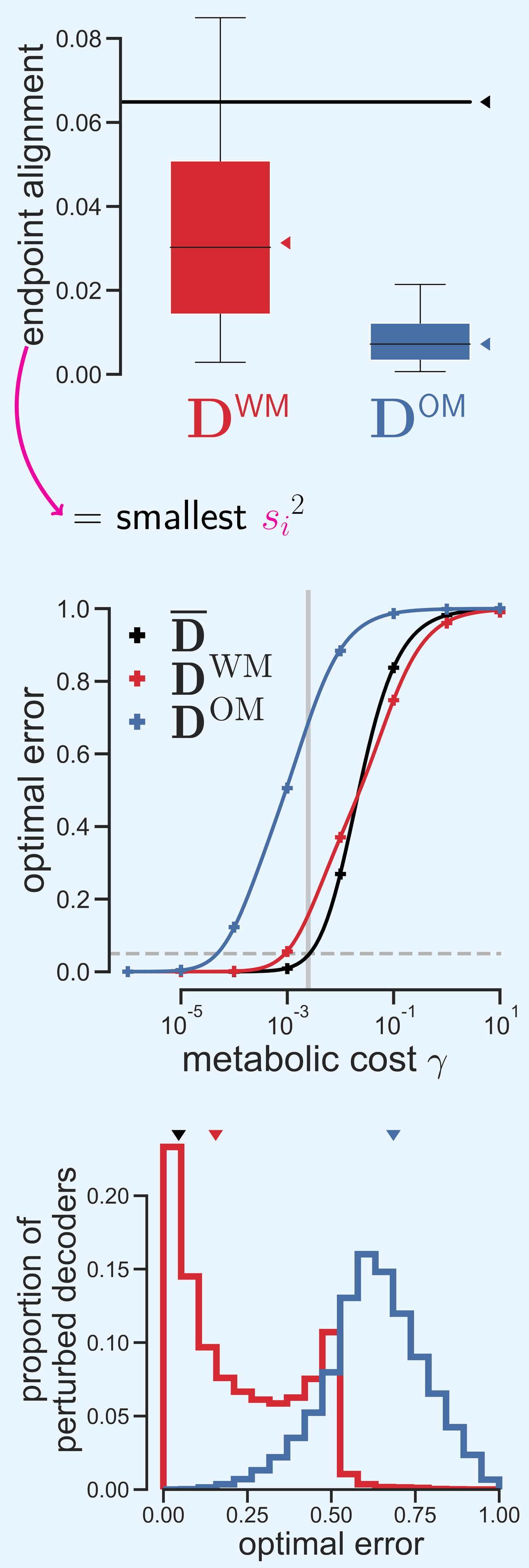
singular values



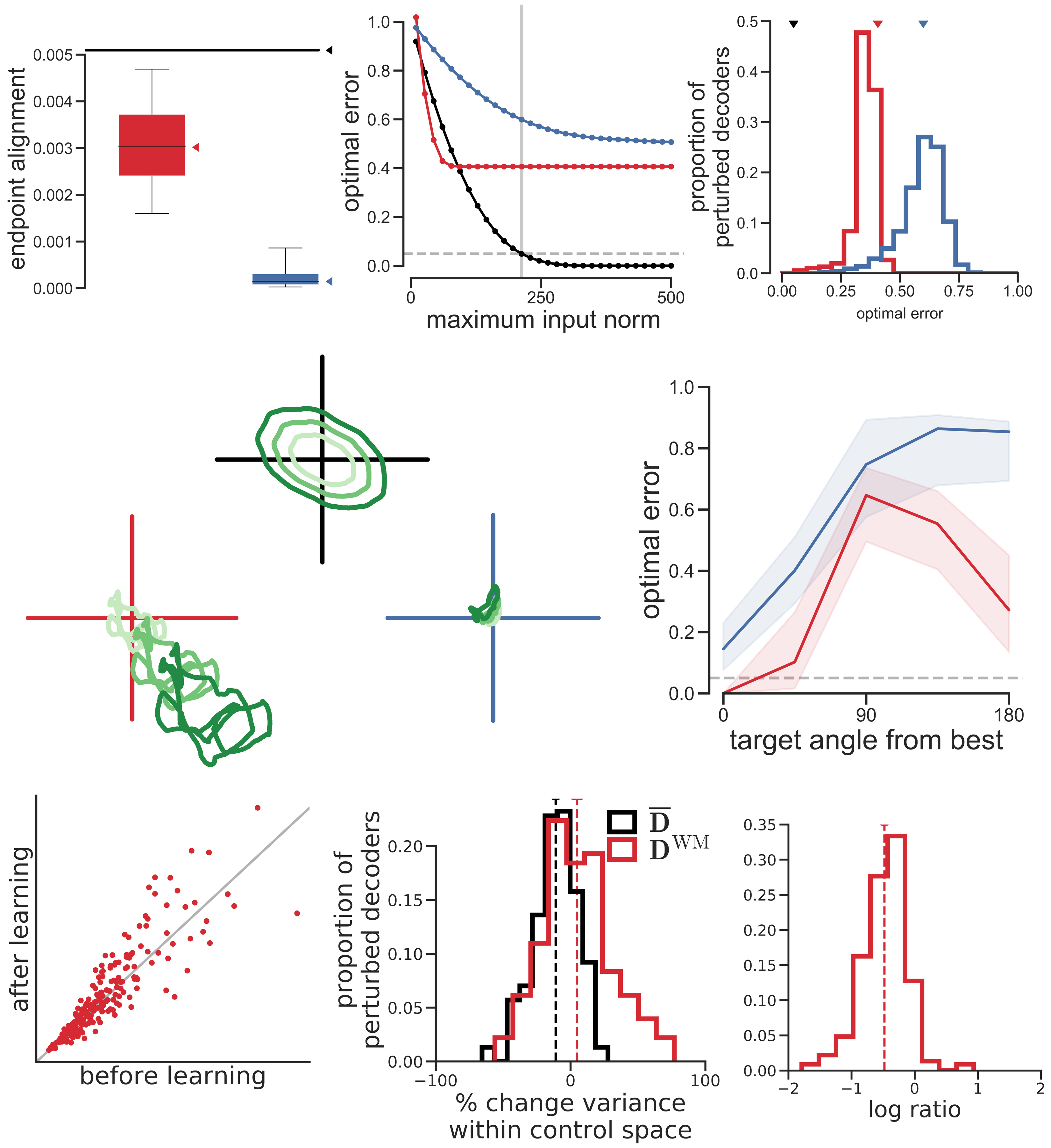
$$\boxed{\langle E_t(\boldsymbol{\theta}^*)\rangle \gtrsim \frac{1}{2}\sum_{i=1}^{2}\frac{1}{(s_i^2/\gamma + 1)^2}}$$

Fast learning by re-aiming

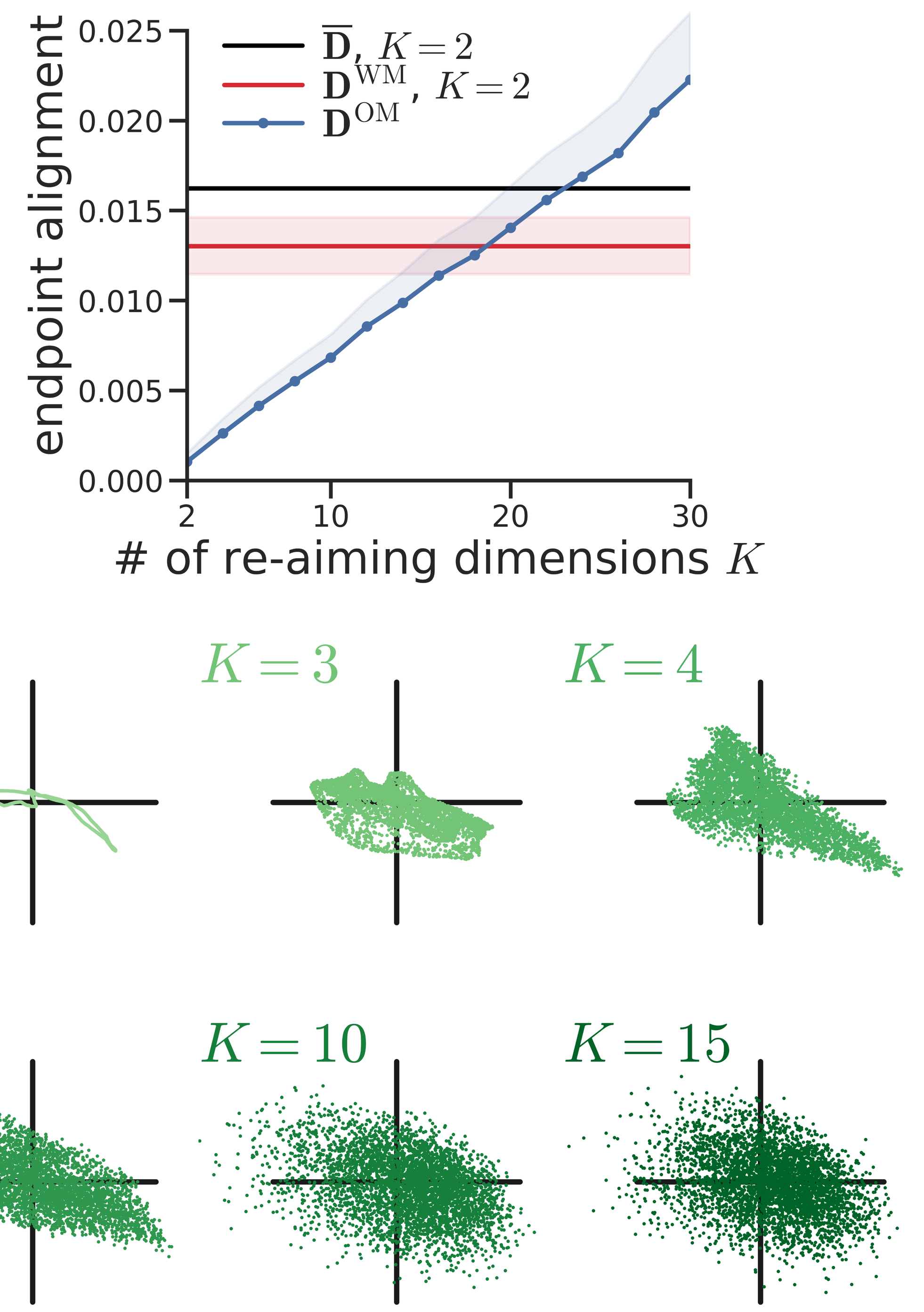
Linear Network



ReLU Network



Slow learning by re-aiming



References:

1. Sadtler, Patrick T., et al. "Neural constraints on learning." *Nature* 512.7515 (2014): 423.
2. Gilja, Vikash, et al. "A high-performance neural prosthesis enabled by control algorithm design." *Nature neuroscience* 15.12 (2012): 1752.
3. Golub, Matthew D., et al. "Learning by neural reassociation." *Nature neuroscience* 21.4 (2018): 607.
4. Hennig, Jay A., et al. "Constraints on neural redundancy." *Elife* 7 (2018): e36774.
5. Ganguly, Karunesh, et al. "Reversible large-scale modification of cortical networks during neuroprosthetic control." Nature neuroscience 14.5 (2011): 662.

This work was funded by a UCL Graduate Research Scholarship and Overseas Research Scholarship.

Contact: jorge.menendez.15@ucl.ac.uk