Decoder Performance Classified as a Function of Neurons

Evan Weiss[1,2], DT Badley[2,6], Erinn Grigsby[2,3,4], Aaron Batista[2,3,4,5,*]

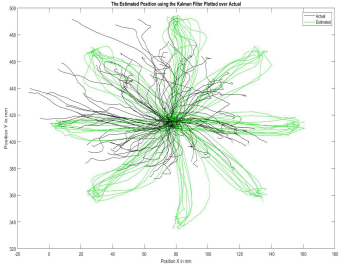
[1]Arizona State University, [2]Center for the Neural Basis of Cognition, Pittsburgh, [3]Dept. of Bioengineering, Univ. Pittsburgh, [4]Systems Neuroscience Center, Univ. Pittsburgh, [5]University of Pittsburgh Brain Institute, Pittsburgh, [6]Worcester Polytechnic Institute , *: denotes equal contributions

Motivation

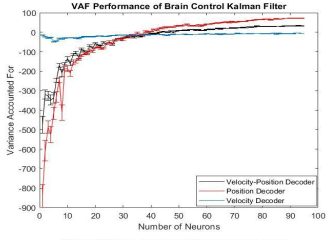
-Brain-Computer Interface (BCI) systems connect neural activity to the control of an external device.
- It stands to reason that the better the neural signals, the better the device control will be.

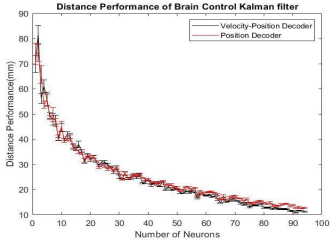
- We sought to characterize the precise relationship between neural signal quality and BCI system performance.
-We expected that as the amount of neurons decrease, decoder performance will decline proportionally. But we wanted to know
(1) What is the shape of that relationship? It could be a sharp decline, or gradual. And (2) Are there "special neurons" that provide more decode performance than others? (3) Would a Kalman Filter or Linear regression decoder be more robust to this change, and similarly, would a position, velocity, or position-velocity decoder fair better?

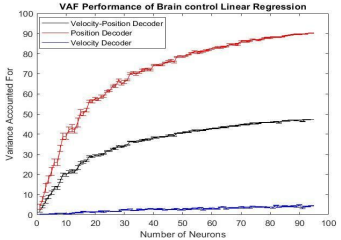
Decoder trajectories against actual values



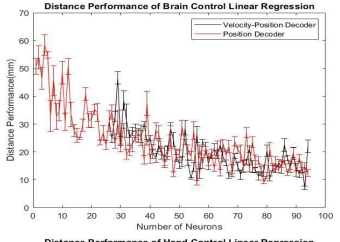
Decoder performance as a function of neurons present

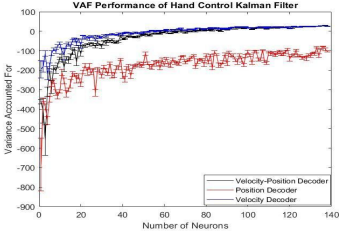


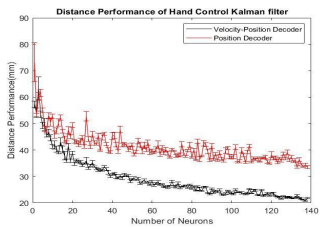


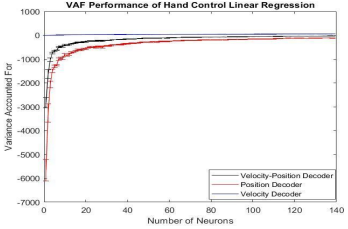


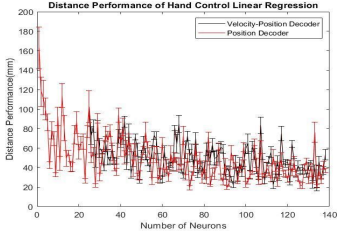
Note: the velocity curves may contain uncertainties that we are looking into.











Performance Metrics

$$VAF\ \% = 100\,\% \times \left(1 - \frac{E\left[(y(t) - \hat{y}(t))^2\right]}{E\left[y(t)^2\right]}\right)$$

$$ReachTargetError = \frac{\left(\sum\left(\frac{\left(\sum|(x_{tg} - \hat{x}_{tg})|\right)}{n} + \frac{\left(\sum|(x_{ty} - \hat{x}_{ty})|\right)}{n}\right)\right)}{N}$$

$\hat{x}_{tg}$ = predicted position
x_{tg} = actual position
$\hat{x}_{ty}$ = predicted position
x_{ty} = actual position
n = number of bins
N = number of trials

$\hat{x}_{tg}$ = predicted position
x_{tg} = actual position
$\hat{x}_{ty}$ = predicted position
x_{ty} = actual position
n = number of bins
N = number of trials

High performance neurons vs indifferent neurons (Variance accounted for)

Brain Control

	Perf. Neurons	Indiff. Neurons
Kal. Velo-Pos	-44.3	24.4
Kal. Pos	-20.98	61.31
Kal. Velo	-21.65	-8.68
Lin. Velo-Pos`	32.56	44.33
Lin. Pos	65.23	84.55
Lin. Velo	0.6158	1.64

Hand Control

	Perf. Neurons	Indiff. Neurons
Kal. Velo-Pos	-79.64	28.15
Kal. Pos	-178.4	-101.83
Kal. Velo	-54.65	23.1
Lin. Velo-Pos`	-269.11	-57.26
Lin. Pos	-555.42	-163.01
Lin. Velo	26.41	47.86

References

1. Ajiboye et al., IEEE, 2010
2.Degenhart et al., Nature Biomedical Engin., 2020
3.Pandarinath et al., eLife, 2017
4.Gilja et al., Nat. Neuroscience, 2012

Conclusions

-We hoped to see the shape of the relationship between number of neurons and performance, to see if "performance" neurons were effective in building decoders, and test to see the robustness of different Kalman Filters and Linear Regression decoders as neurons are reduced..
-After conducting our neuron-reducing analysis, performance behaved in an exponential fashion, with the position decoder performing the best on average.
-Our hypothesis that high performance neurons would perform better than indifferent neurons was disproved.
-By the data, it appears that there is a casual relationship between neuron number and performance.

Future Questions

-Does the order when a performance neuron is dropped matter? Also, perform the same analysis using PVA & OLE decoders
-Perform neuron-dropping in a manner that respects or ignores the low-dimensional "neural manifold" and see if the outcome is different.