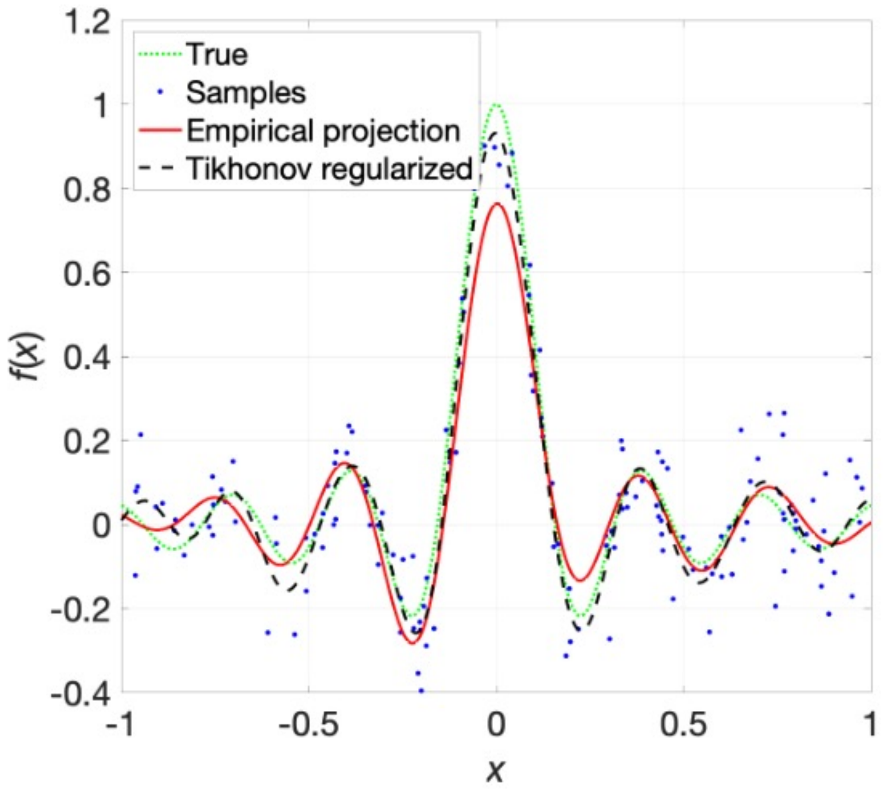
Computing on Functions Using Randomized Vector Representations

E. Paxon Frady, Denis Kleyko, **Christopher J. Kymn***,
Bruno A. Olshausen, Friedrich T. Sommer

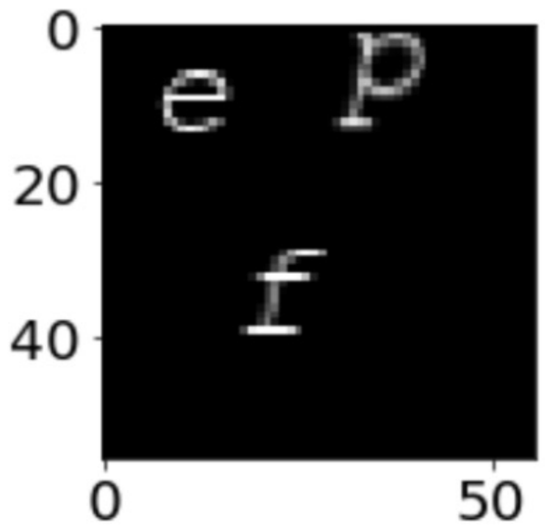
9th Annual Neuro Inspired Computational Elements Workshop, March 29, 2022

*cjkymn@berkeley.edu

A framework for computing over distributed representations, with connections to…

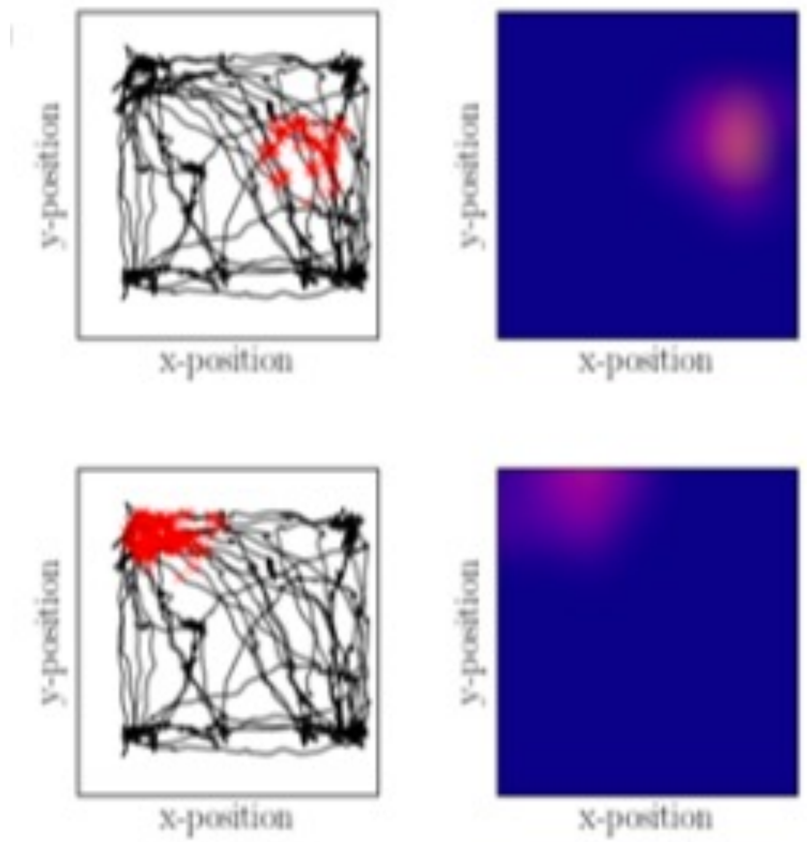


Kernel methods in machine learning



$$
\begin{aligned}
\mathbf{v}_{scene} &= \mathbf{v}_e \odot \mathbf{x}^{12} \odot \mathbf{y}^{10} \\
&+ \mathbf{v}_p \odot \mathbf{x}^{35.86} \odot \mathbf{y}^{7.22} \\
&+ \mathbf{v}_f \odot \mathbf{x}^{22.7} \odot \mathbf{y}^{35}
\end{aligned}
$$

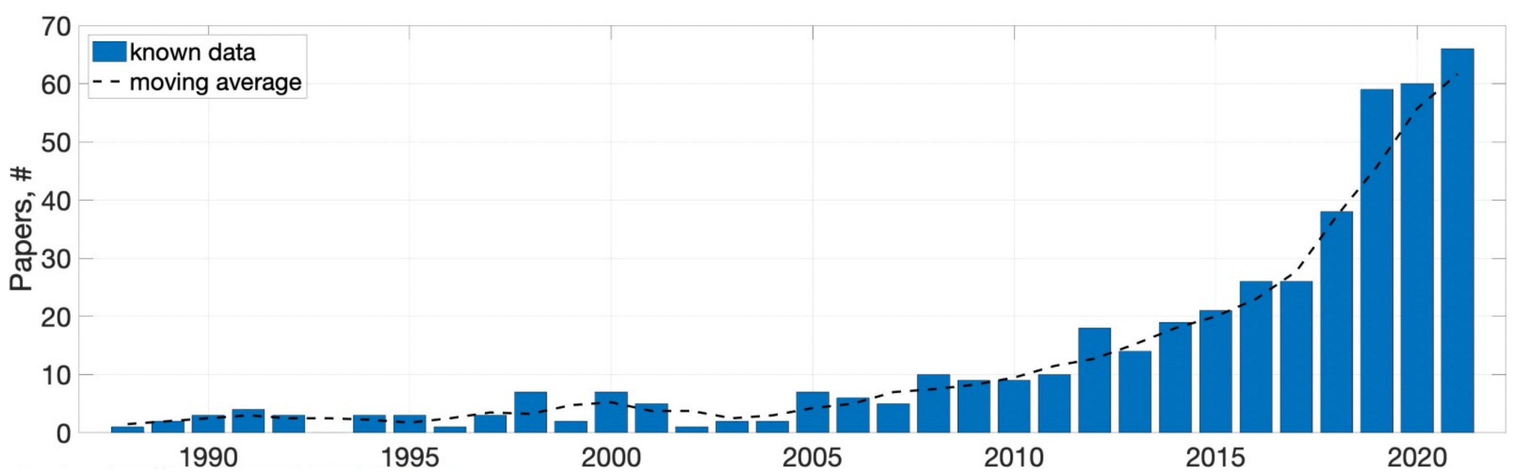
Factorizable image encodings



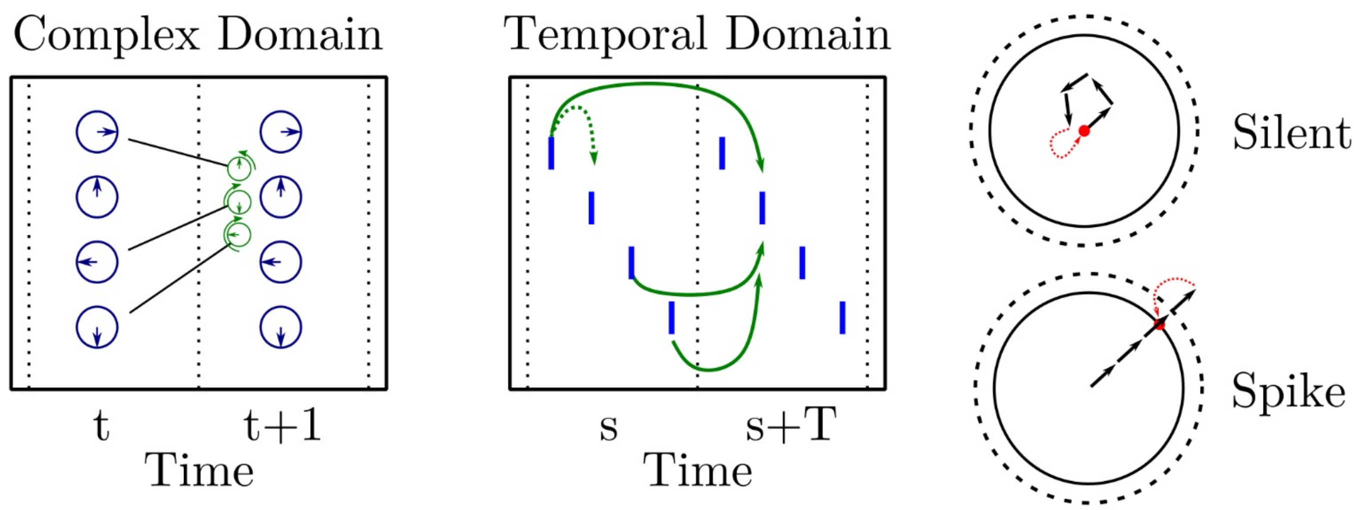
Modeling in neuroscience

Our work highlights and extends work in Vector Symbolic Architectures (VSA) / Hyperdimensional Computing (HDC)

VSA/HDC related papers per year



Robust computation with rhythmic spike patterns



Frady, E.P., Sommer, F.T. (2019) Robust computation with rhythmic spike patterns. PNAS 116(36) 18050-59.

Vector Symbolic Architectures provide a principled framework for computing with distributed representations

$a, b, c \ldots = V^N$
(i.i.d. random vectors)

$a^T b$: *similarity*
(inner product)

$a + b$: *superposition*
(vector sum)

$\boxed{a \odot b : \textit{binding operation} \text{ (element-wise multiplication)} \quad !}$

$\rho(a)$: *ordering operation*
(cyclic shift, permutation)

Binary	Bipolar	Real	Complex
Binary spatter code (Kanerva, 1996) Binary sparse distributed code (Rachkovskij, 2001)	Multiply, Add, Permute (Gayler, 1998) Hyperdimensional Computing (Kanerva, 2009)	Holographic Reduced Representation (Plate, 1991) Matrix binding with additive terms (Gallant & Okaywe, 2013)	Fourier Holographic Reduced Representations (Plate, 2003)

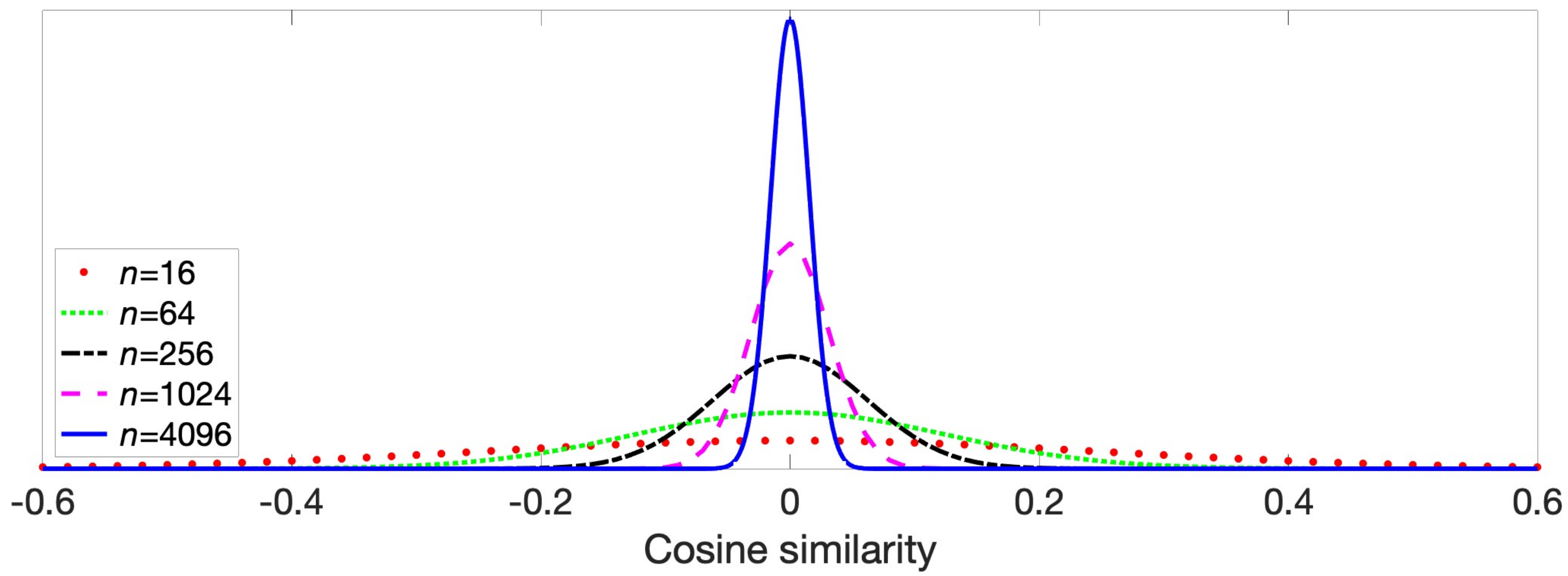
A kernel perspective on symbolic VSA

Maximum separation of symbols with orthogonal representations:

$$\mathbf{z}(s_1)^\top \overline{\mathbf{z}(s_2)} \xrightarrow{\text{large } n} K_{Kron}(s_1, s_2) := \delta_{s_1,s_2}$$

Good separation with i.i.d. pseudorandom representations:

$$\left| \mathbf{z}(s_1)^\top \overline{\mathbf{z}(s_2)} - K_{Kron}(s_1, s_2) \right| \leq \epsilon(n)$$



VSA can represent data structures with high-dimensional vectors

Hash Table:

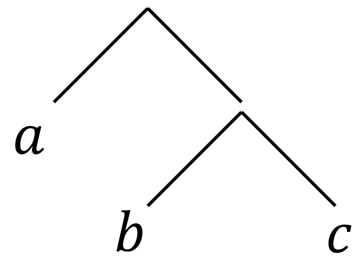
$$s = x \odot a + y \odot b + z \odot c$$

Sequence:

$$s = k \odot a + \rho(k) \odot b + \rho^2(k) \odot c$$

Tree:

$$s = l \odot a + r \odot \rho(l \odot b) + r \odot \rho(r \odot c)$$



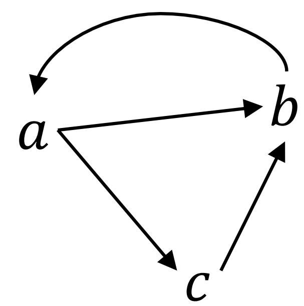
2-D Space/Matrix:

$$s = x \odot y \odot a + x \odot \rho(y) \odot b + \rho(x) \odot y \odot c$$

a	c
b	

Graph:

$$s = a \odot \rho(b) + a \odot \rho(c) + c \odot \rho(b) + b \odot \rho(a)$$



Kleyko, D., Davies, M., Frady, E. P., Kanerva, P., Kent, S. J., Olshausen, B. A., ... & Sommer, F. T. (2021). Vector symbolic architectures as a computing framework for nanoscale hardware. *arXiv preprint arXiv:2106.05268*.

Kernel locality preserving encoding (KLPE)

A randomizing encoding function:

$$f : r \in \mathbb{R} \to \mathbf{z}(r) \in \mathbb{C}^n$$

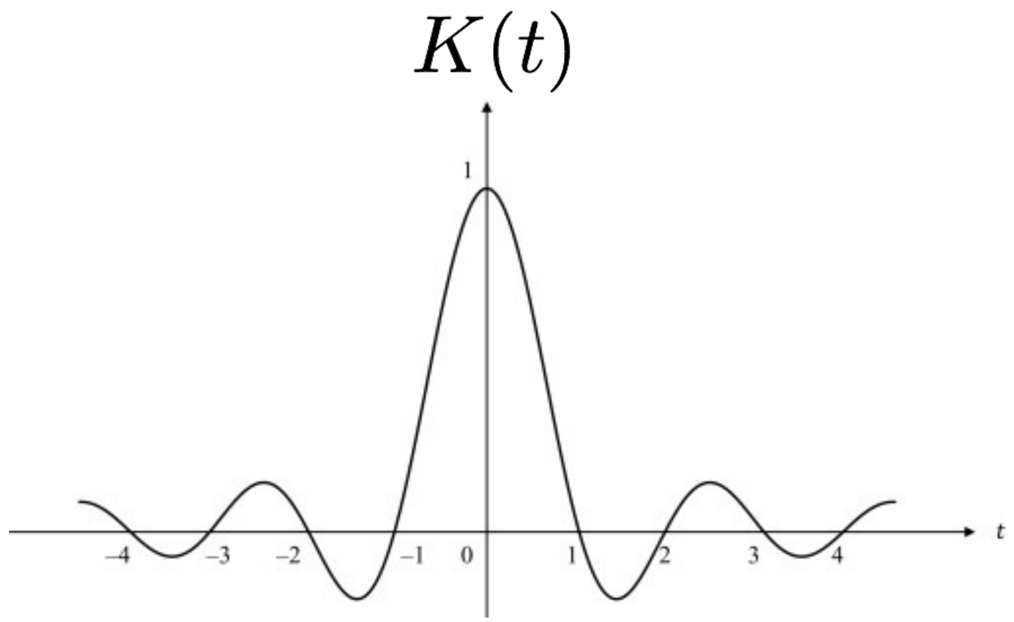
such that:

(i) Inner product forms a kernel:

$$\mathbf{z}(r_1)^\top \overline{\mathbf{z}(r_2)} \xrightarrow{\text{large } n} K(r_1 - r_2)$$

(ii) Translation is computed by binding:

$$\mathbf{z}(r_1 + r_2) = \mathbf{z}(r_1) \circ \mathbf{z}(r_2)$$



Important Results from Functional Analysis

(1) Inner product kernels define a reproducing kernel Hilbert space:

Definition: A kernel $K(\mathbf{x}_1, \mathbf{x}_2)$ is positive definite if, for any finite set of points $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$, the Gram matrix $K(\mathbf{x}_i, \mathbf{x}_j)$ is positive definite (i.e., all eigenvalues are non-negative).

Theorem: All inner product kernels are positive definite (Schölkopf et al., 2002; Hofmann et al., 2008).

Theorem (Aronszajn, 1950): Each positive definite kernel defines a reproducing kernel Hilbert space (RKHS).

(2) The kernel shape is defined by the Fourier transform of a probability distribution:

Theorem (Bochner, 1932): Each continuous kernel $K(\mathbf{x} - \mathbf{y})$ is positive definite if, and only if, it is the Fourier transform of a positive definite measure $p(\boldsymbol{\omega})$:

$$K(\mathbf{x} - \mathbf{y}) = \int d\boldsymbol{\omega}\; p(\boldsymbol{\omega}) e^{\imath\, \boldsymbol{\omega}^\top (\mathbf{x} - \mathbf{y})} = E_{p(\boldsymbol{\omega})}\left[e^{\imath\, \boldsymbol{\omega}^\top \mathbf{x}} \overline{e^{\imath\, \boldsymbol{\omega}^\top \mathbf{y}}} \right] \tag{5}$$

Vector Function Architecture (VFA = VSA + KLPE)

The function:
$$f(r) = \sum_k \alpha_k K(r - r_k)$$

Is represented by the vector:
$$\mathbf{y}_f = \sum_k \alpha_k \mathbf{z}(r_k)$$

FPE with Hadamard product binding:

$\mathbf{z}^{hp}(r) := (\mathbf{z})^r$ (complex-valued)

FPE with Circular convolution binding:

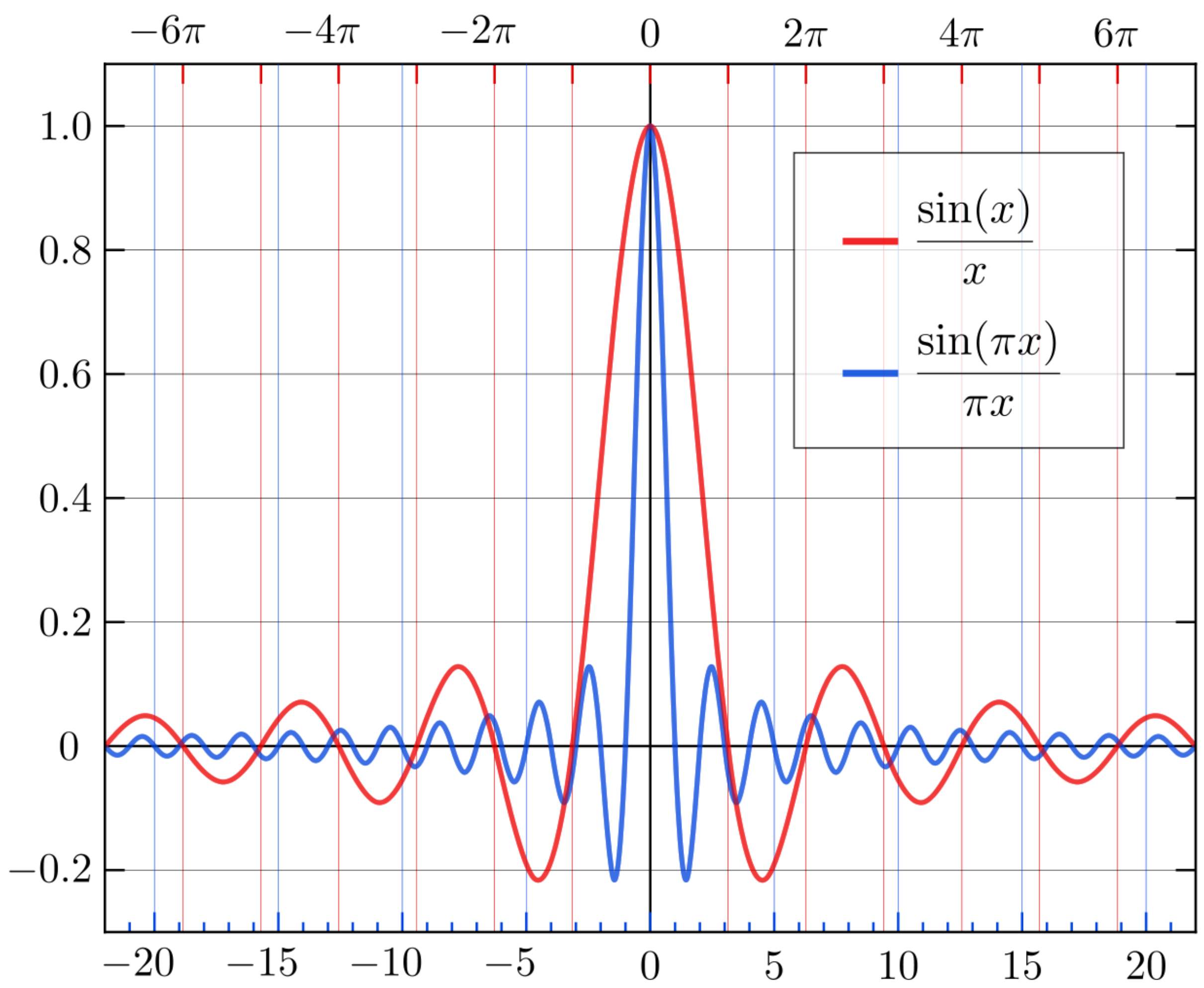
$\mathbf{z}^{cc}(r) := \mathbf{z}^{(\circledast r)} = \mathcal{F}^{-1}\left(\mathcal{F}(\mathbf{z})^r\right) = F^{-1}\left(F\mathbf{z}\right)^r$ (real-valued)

Block-local circular convolution:

$\mathbf{z}^{lcc}(r)_{(\text{block } i)} := \mathbf{z}^{(*_B\, r)}_{(\text{block } i)} = F^{-1}\left(F\mathbf{z}_{(\text{block } i)}\right)^r$ (sparse)

VFAs with uniformly sampled base vectors result in a sinc kernel

Theorem 2: Assume an FPE with a uniformly sampled base vector, which is the typical procedure for sampling VSA vectors. For a Hadamard FPE, this means the phases of the base vector are sampled from the uniform phase distribution. The FPE then induces a VFA which is the RKHS of band-limited continuous functions, independent of the underlying realization of the binding operation. Specifically, the kernel of FPE is the sinc function, which defines the RKHS of the band-limited continuous functions.



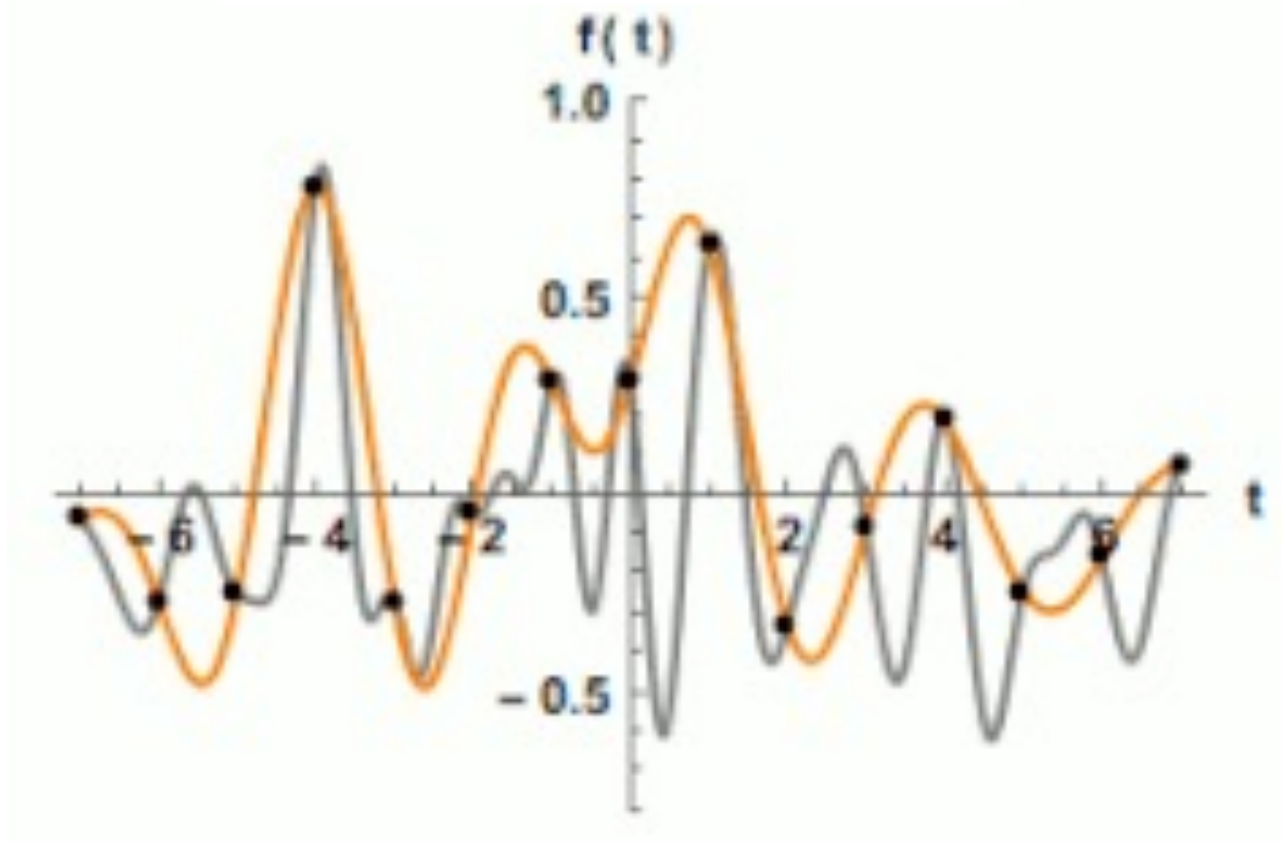
https://en.wikipedia.org/wiki/Sinc_function

Approximating functions with sinc

$$x(t) = \sum_{n=-\infty}^{\infty} x[n]\,\mathrm{sinc}\left(\frac{t-nT}{T}\right)$$

Function Representation in VFA

$$f(r) = \sum_k \alpha_k K(r - r_k)$$



https://en.wikipedia.org/wiki/Whittaker%E2%80%93Shannon_interpolation_formula

Manipulating functions with VFA

- Point-wise readout of a function

$$f(s) = \langle f, K_s \rangle = \mathbf{y}_f^\top \overline{\mathbf{z}(s)}$$

- Point-wise addition

$$\mathbf{y}_{f+g} = \mathbf{y}_f + \mathbf{y}_g$$

- Function shifting

$$f(x) \to g(x) = f(x+r)$$

$$\mathbf{y}_g = \mathbf{y}_f \circ \mathbf{z}(r)$$

- Function convolution

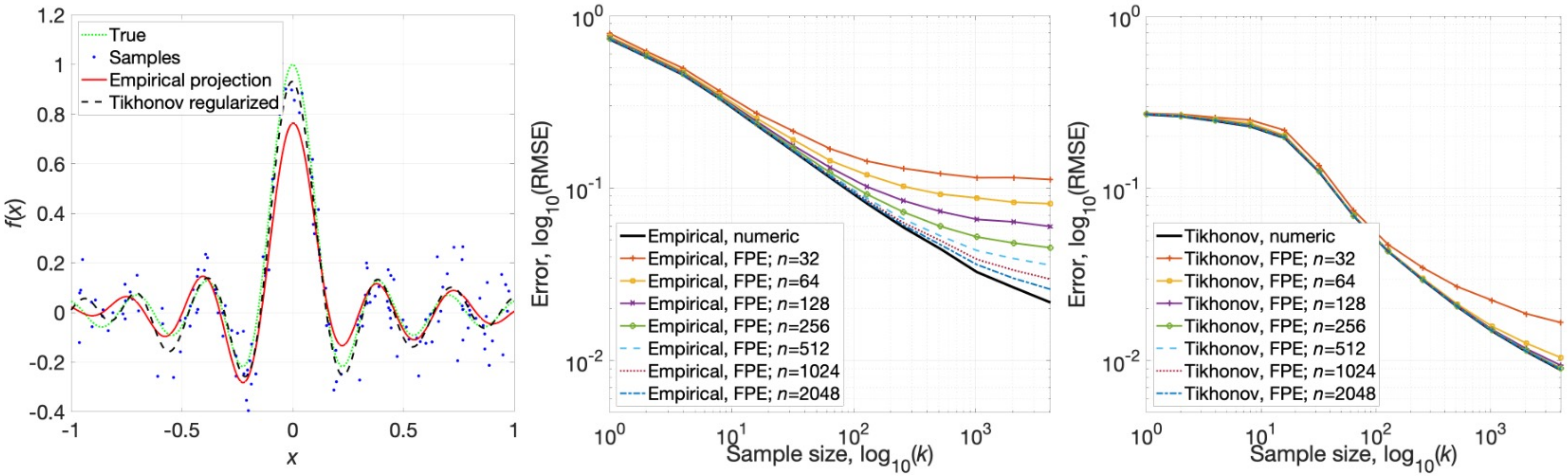
$$\mathbf{y}_{f*g} = \mathbf{y}_f \circ \mathbf{y}_g$$

- Overall similarity between functions

$$\langle f, g \rangle = \mathbf{y}_f^\top \overline{\mathbf{y}_g}$$

Application: regression with sinc kernels

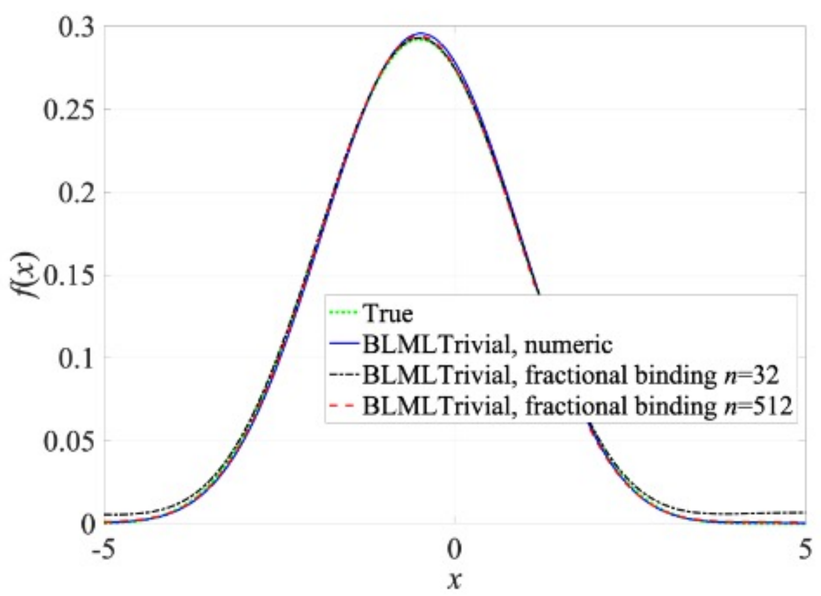
$$\hat{f}_{c,k,n}(x) = \frac{2c}{k\pi}\sum_{i=1}^{k} Y_i \mathrm{sinc}\left(\frac{c}{\pi}(X_i - x)\right) = \frac{2c}{kn\pi}\sum_{i=1}^{k} Y_i \mathbf{z}(\frac{c}{\pi}X_i)^\top \overline{\mathbf{z}(\frac{c}{\pi}x)} = (\mathbf{y}^X)^\top \overline{\mathbf{z}(\frac{c}{\pi}x)},$$

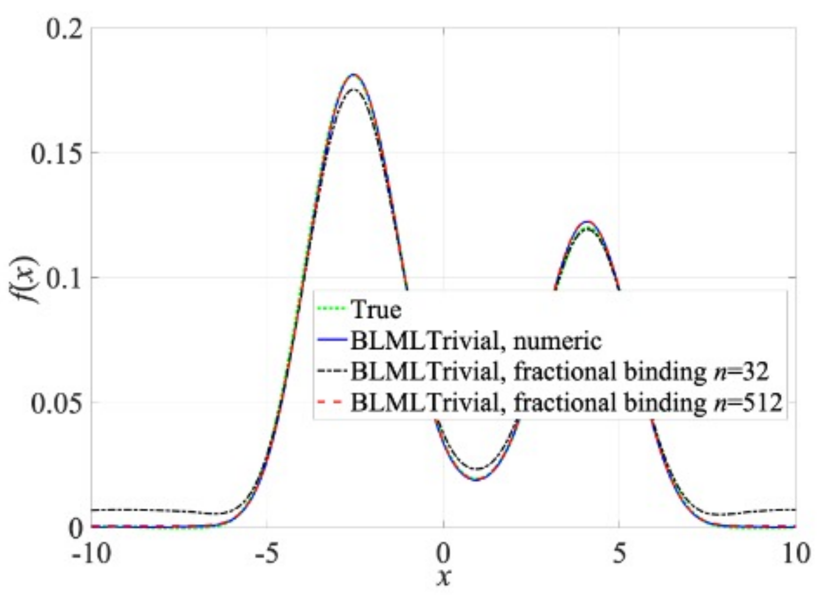


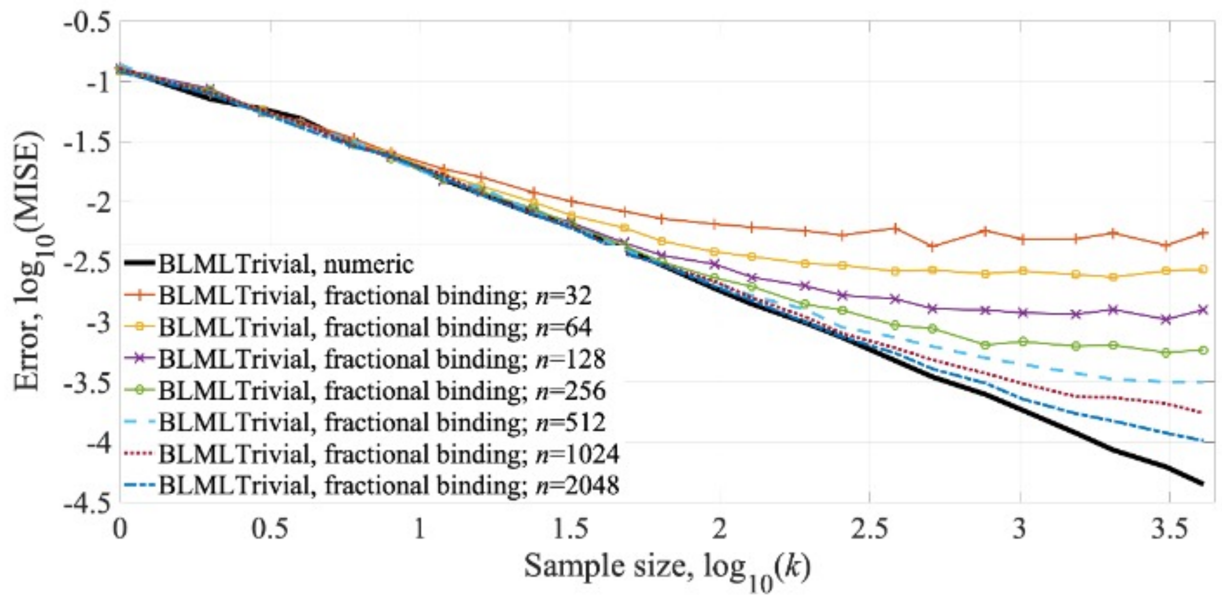
Application: Kernel density estimation in VFA

Density estimation with band-limited functions (Agarwal et al. 2017):

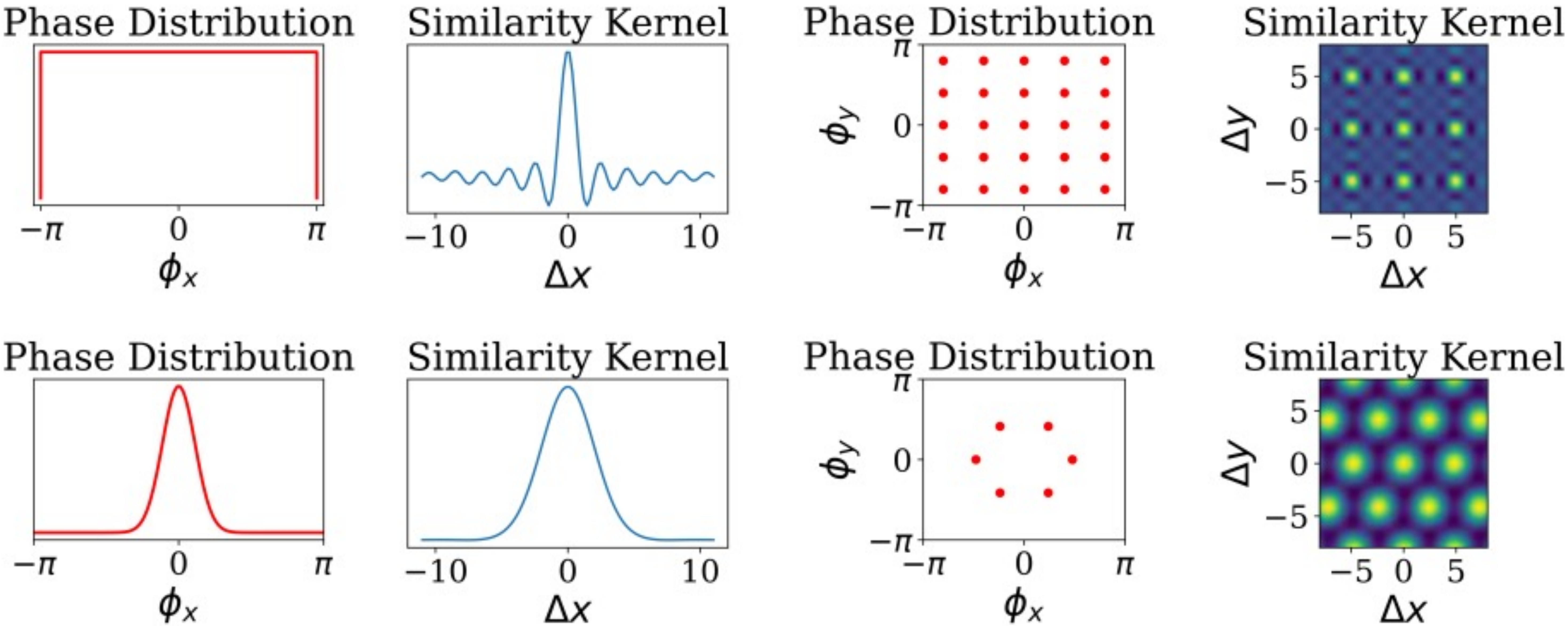
$$p(r) = \left(\frac{f_c}{k} \sum_{i=1}^{k} \hat{c}_i K(f_c(r - r_i)) \right)^2 = \left(\frac{f_c}{kn} \sum_{i=1}^{k} \hat{c}_i \mathbf{z}(f_c r_i) \overline{\mathbf{z}(f_c r)} \right)^2 = \left((\mathbf{y}^p)^\top \overline{\mathbf{z}(f_c r)} \right)^2$$



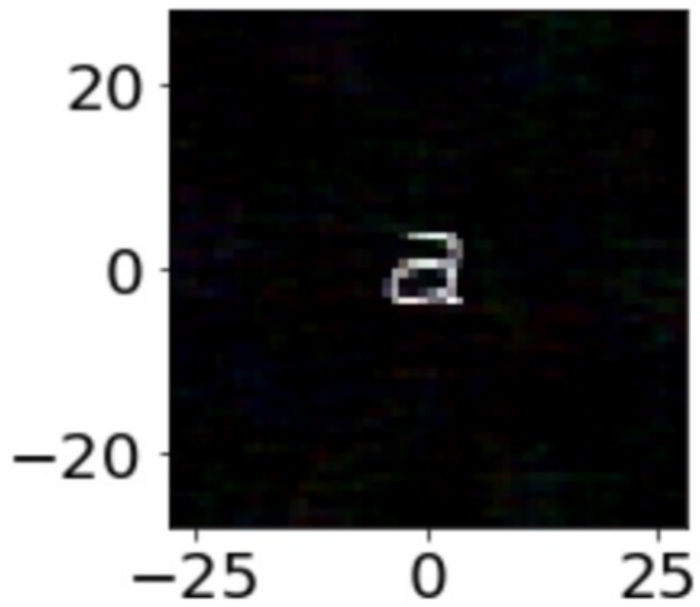




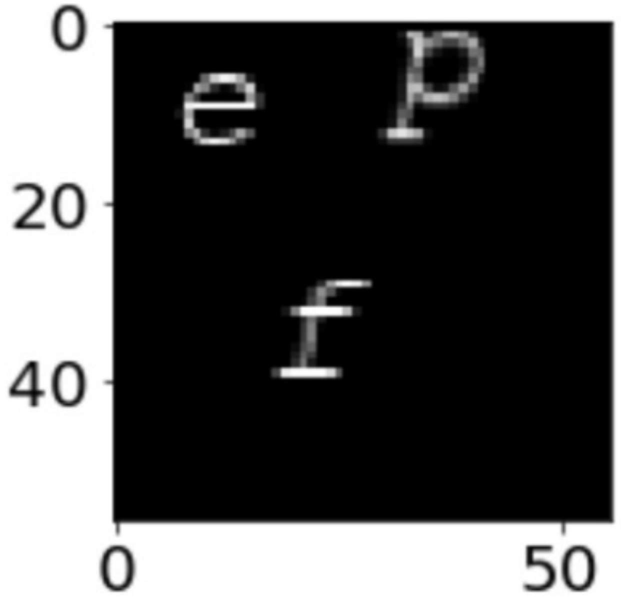
Phase distribution of the base vector determines similarity kernel



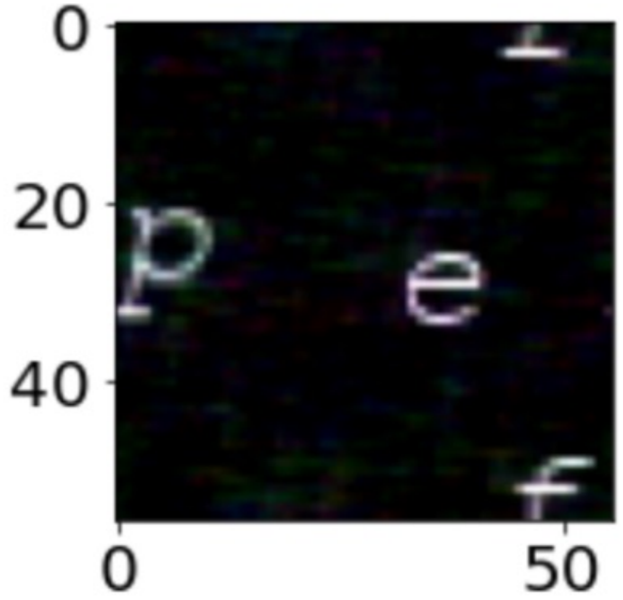
Application: Representing images in VFA



$$\mathbf{v}_a = \sum_{x,y} Im(x,y) \cdot \mathbf{x}^x \odot \mathbf{y}^y$$



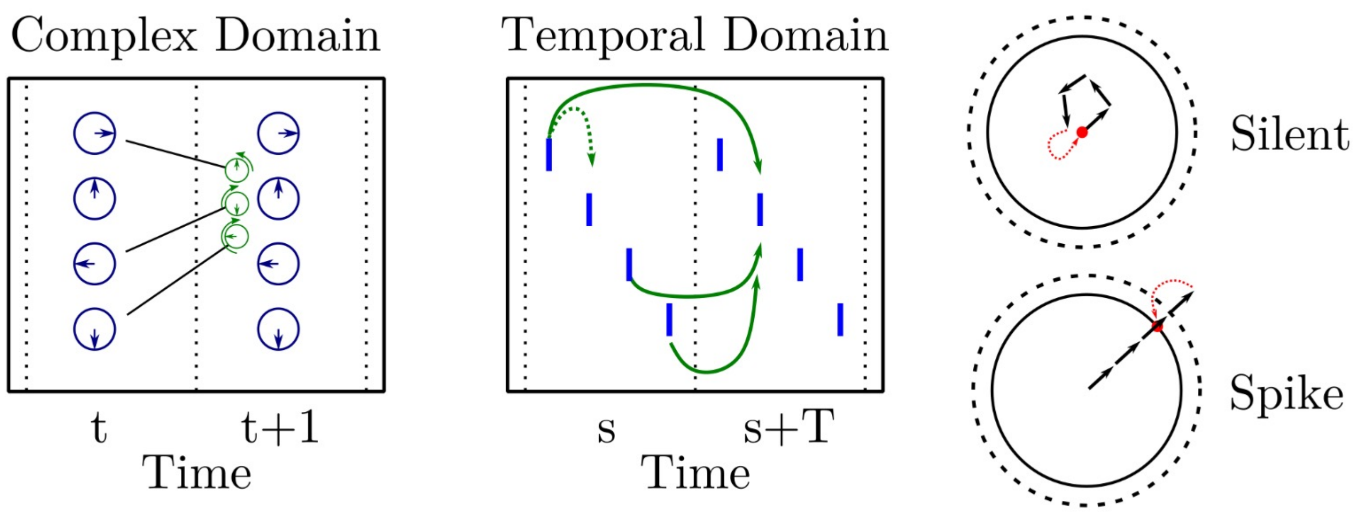
$$\begin{aligned}\mathbf{v}_{scene} &= \mathbf{v}_e \odot \mathbf{x}^{12} \odot \mathbf{y}^{10} \\ &+ \mathbf{v}_p \odot \mathbf{x}^{35.86} \odot \mathbf{y}^{7.22} \\ &+ \mathbf{v}_f \odot \mathbf{x}^{22.7} \odot \mathbf{y}^{35}\end{aligned}$$



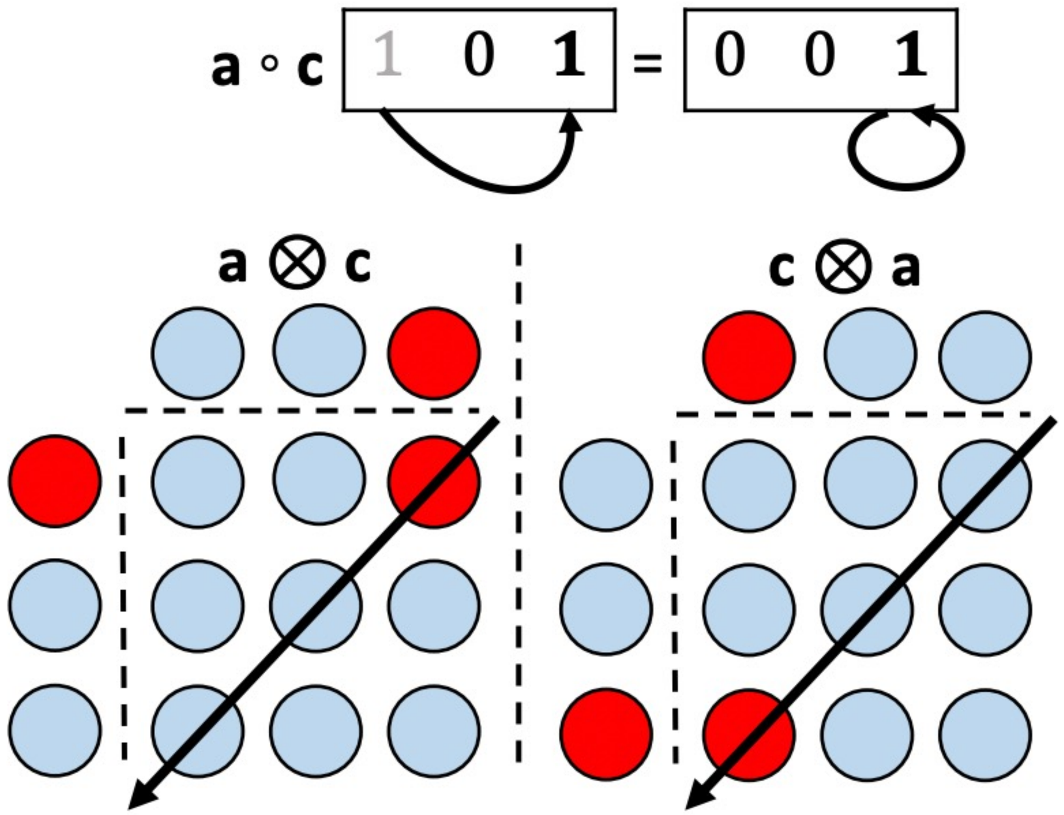
$$\mathbf{v}^{tr}_{scene} = \mathbf{v}_{scene} \odot \mathbf{x}^{25.8} \odot \mathbf{y}^{20.2}$$

Relating VSA principles to neural coding

Mapping complex-valued vectors to spike timing codes

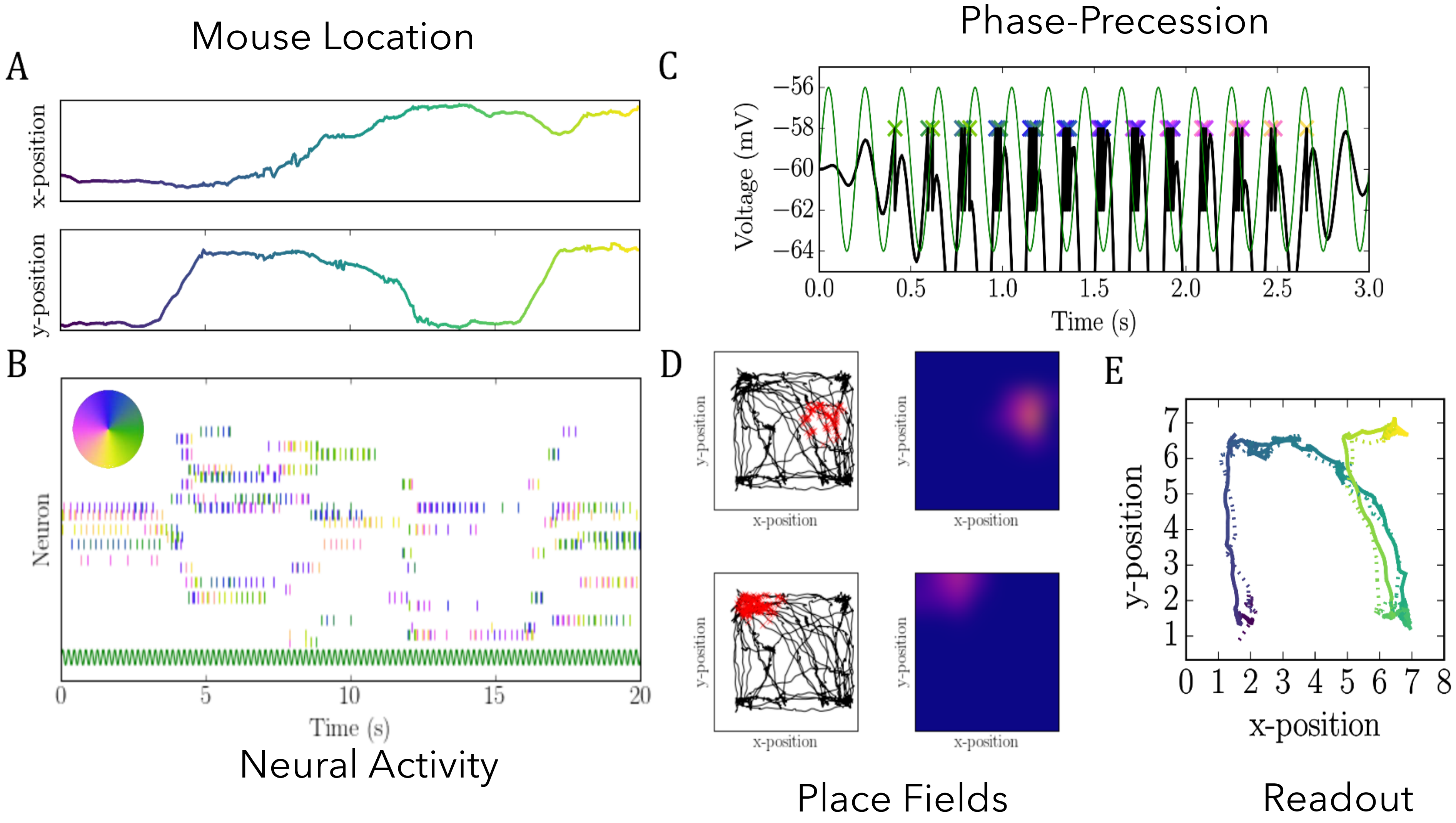


Sparse Block Codes provide a sparse implementation of VSA/VFA



Frady, E.P., Sommer, F.T. (2019) Robust computation with rhythmic spike patterns. PNAS 116(36) 18050-59.
M. Laiho, et al., "High-Dimensional Computing with Sparse Vectors," IEEE Biomedical Circuits and Systems Conference (BioCAS), 2015.
E. P. Frady, et al., "Variable Binding for Sparse Distributed Representations: Theory and Applications," IEEE Transactions on Neural Networks and Learning Systems, 2021.

An encoding model of hippocampus predicts place fields and phase precession



Frady, P., Kanerva, P., & Sommer, F. (2018). A framework for linking computations and rhythm-based timing patterns in neural firing, such as phase precession in hippocampal place cells. In *Proceedings of the Conference on Cognitive Computational Neuroscience*.

Key Takeaways

- A unified framework for reasoning about symbols and real values with distributed representations (extending VSA -> VFA)
- Kernel methods in machine learning can now be integrated with VSA methods.
- Predictions for neural coding principles and single-cell representations (e.g., in hippocampus).