Learning in neuronal networks:

Processing high-order statistics embedded in time series for classification tasks

Matthieu Gilson

Chaire of Junior Prof

https://matthieugilson.eu

- Sofia Lawrie
- Rubén Moreno-Bote

upf.
Universitat
Pompeu Fabra
Barcelona

- Sandra Nestler
- David Dahmen
- Moritz Helias

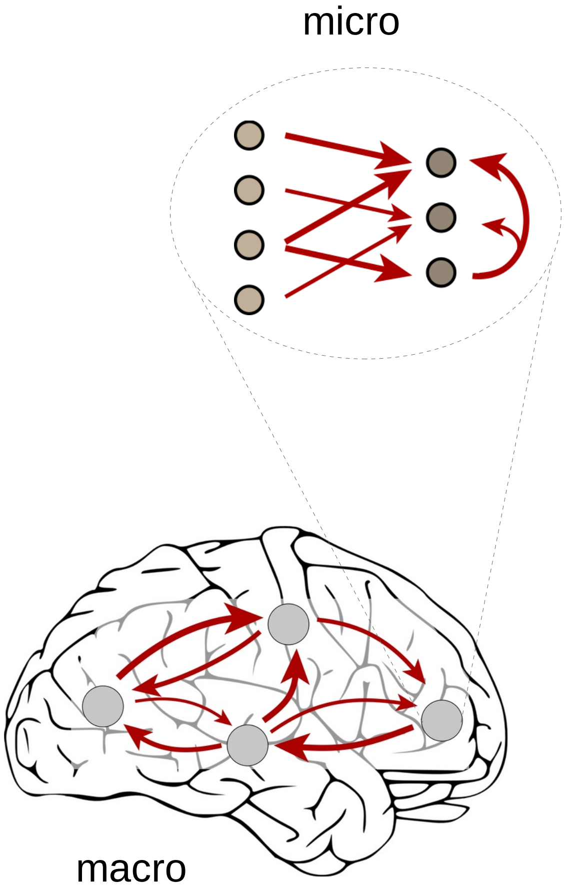
JÜLICH
Forschungszentrum

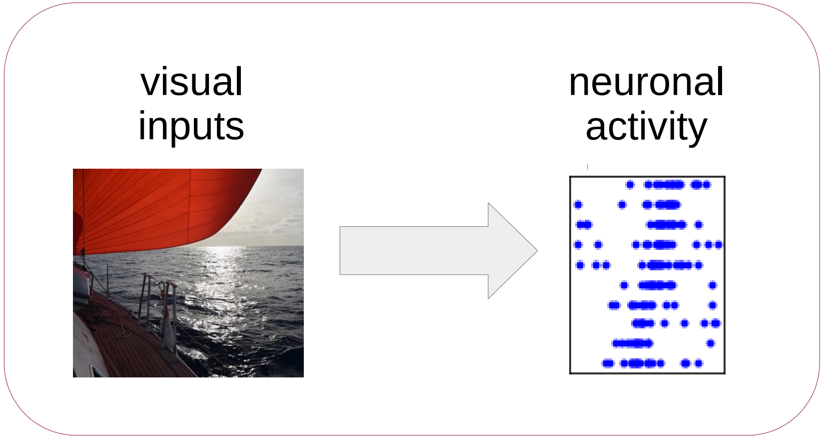
- Laurent Perrinet

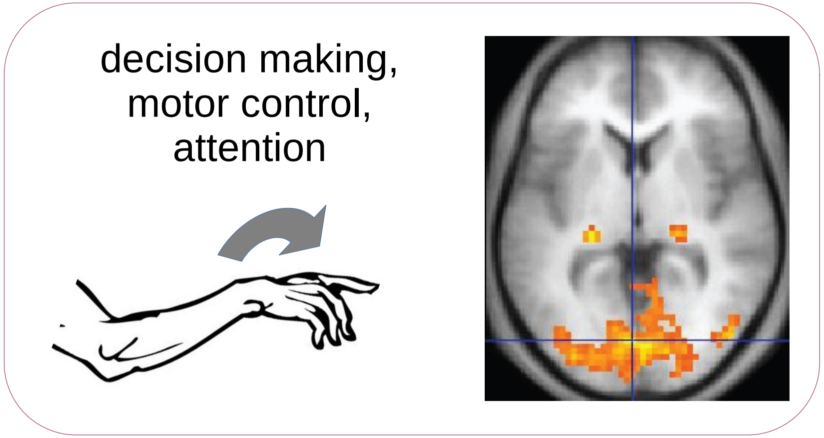
int
institut de
neurosciences
de la timone

Institut de
Neurosciences des
Systèmes

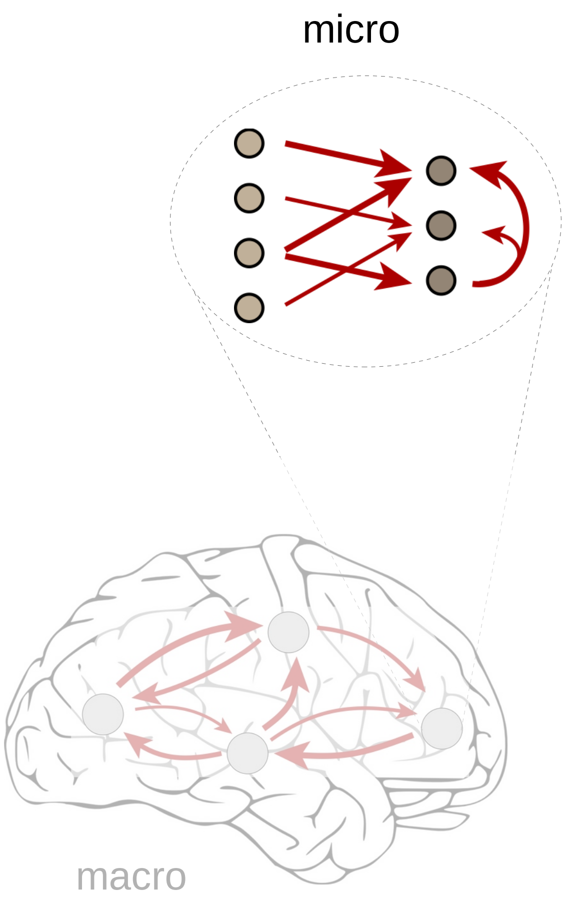
Neuronal processing in the brain: computations and communication

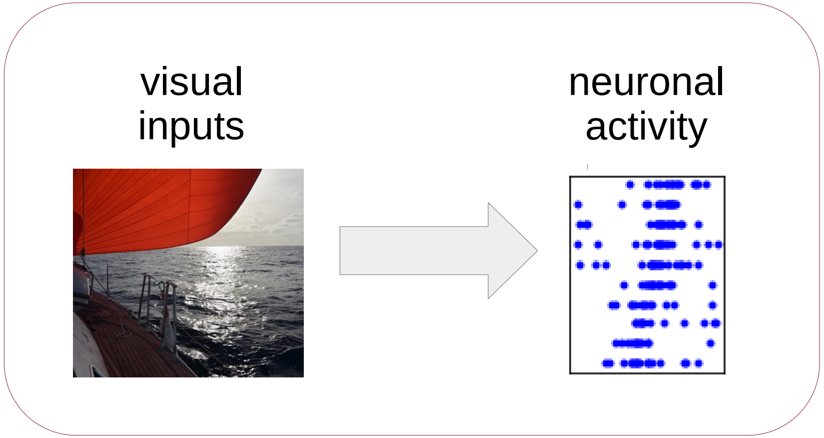


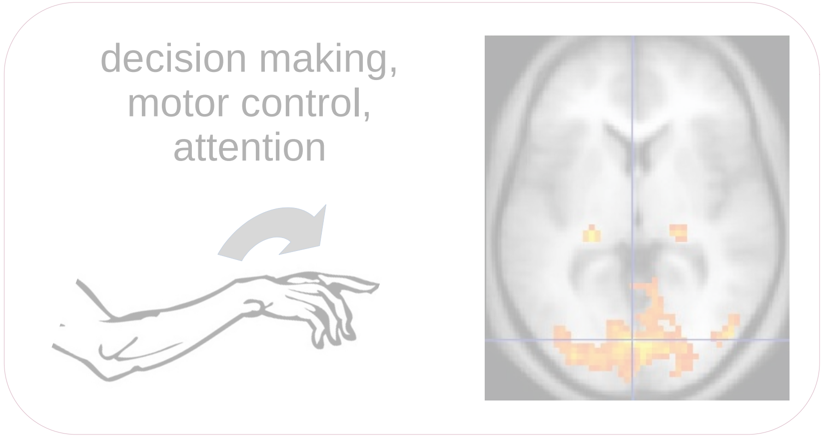




Neuronal processing in the brain: computations and communication

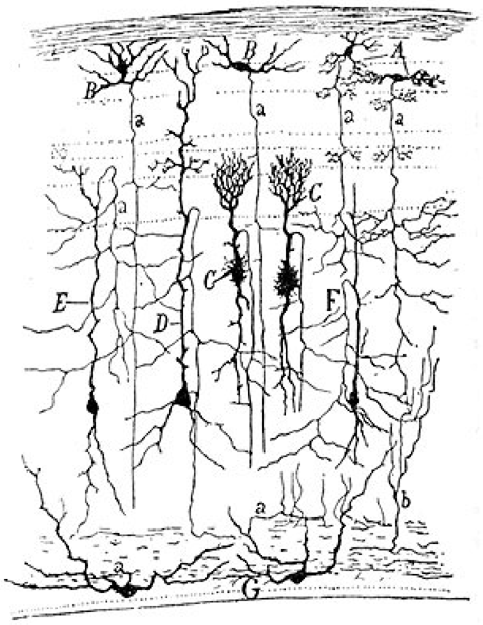






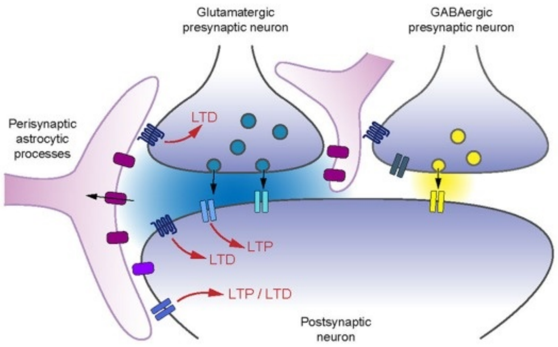
How does the brain implement functions?

- Karl Lashley (1890-1958): storage of memory in brain regions (engram)



Ramon y Cajal 1905

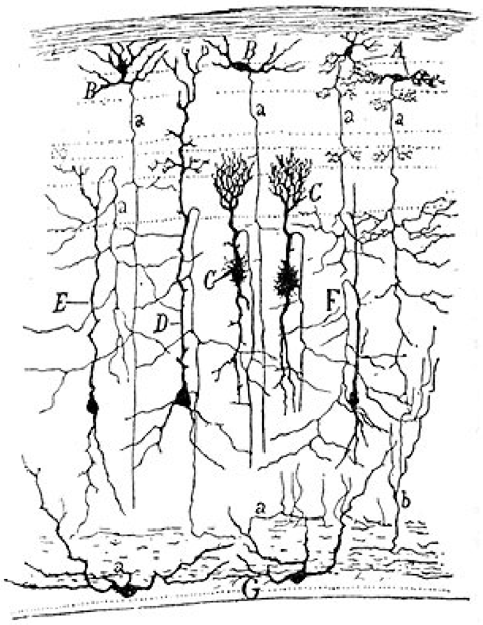
- Synaptic plasticity (learning) shapes the neuronal network dynamics



Valcheva et al. *Front Syn Neurosci* 2019

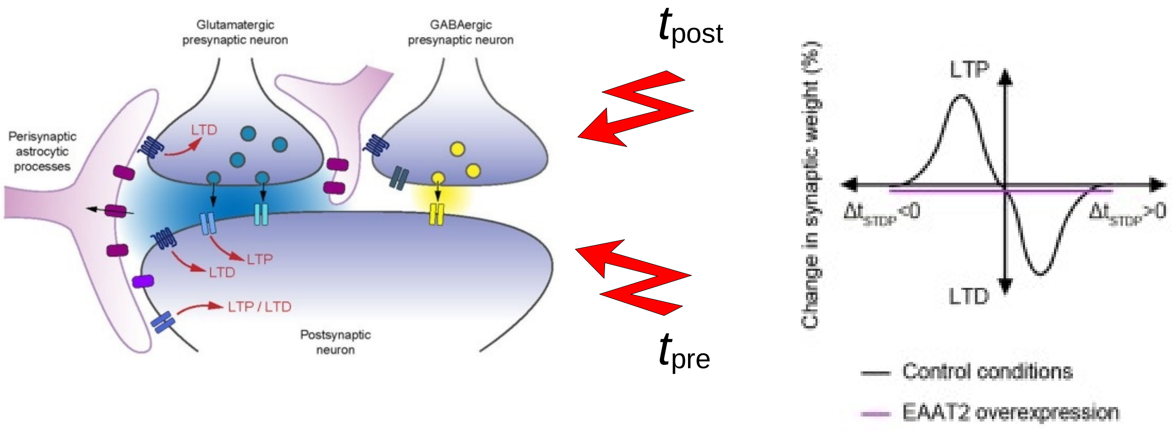
How does the brain implement functions?

- Karl Lashley (1890-1958): storage of memory in brain regions (engram)



Ramon y Cajal 1905

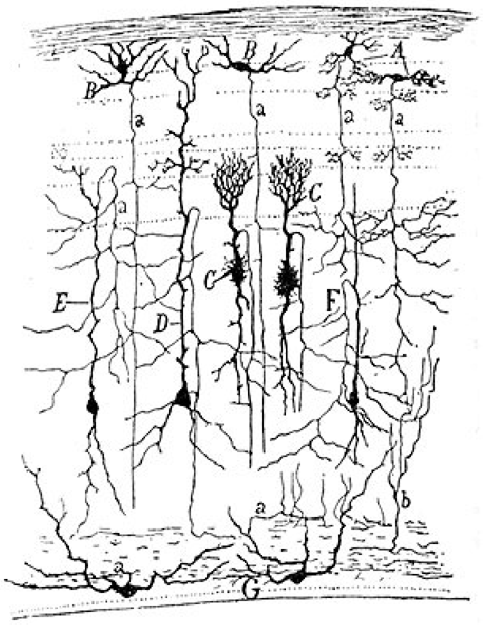
- Synaptic plasticity (learning) shapes the neuronal network dynamics



Valcheva et al. *Front Syn Neurosci* 2019

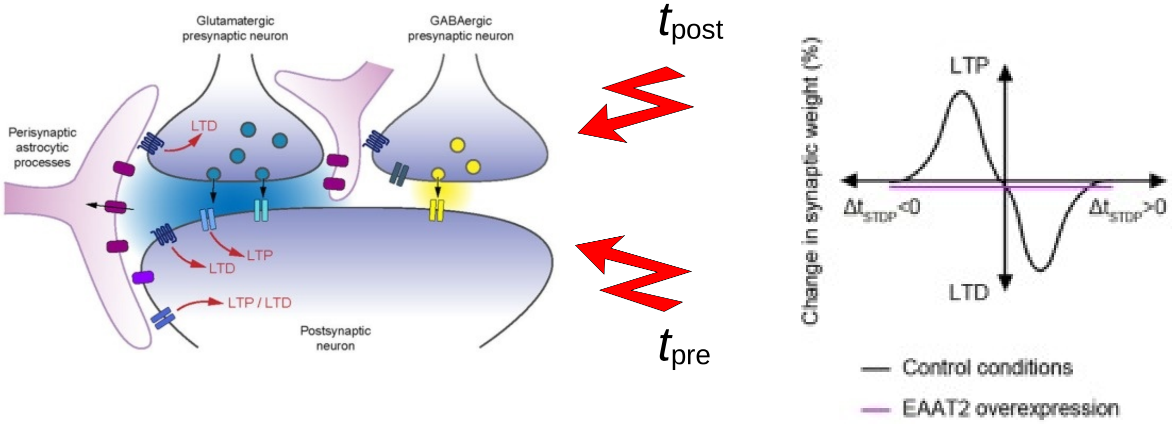
How does the brain implement functions?

- Karl Lashley (1890-1958): storage of memory in brain regions (engram)



Ramon y Cajal 1905

- Synaptic plasticity (learning) shapes the neuronal network dynamics

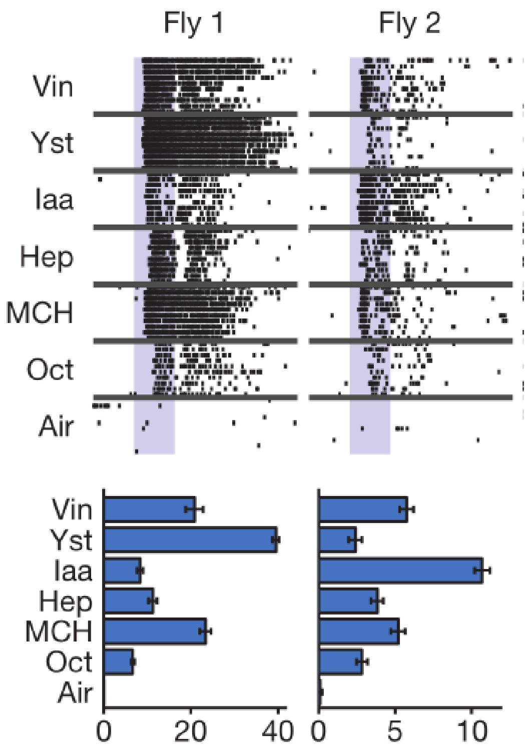


Valcheva et al. *Front Syn Neurosci* 2019

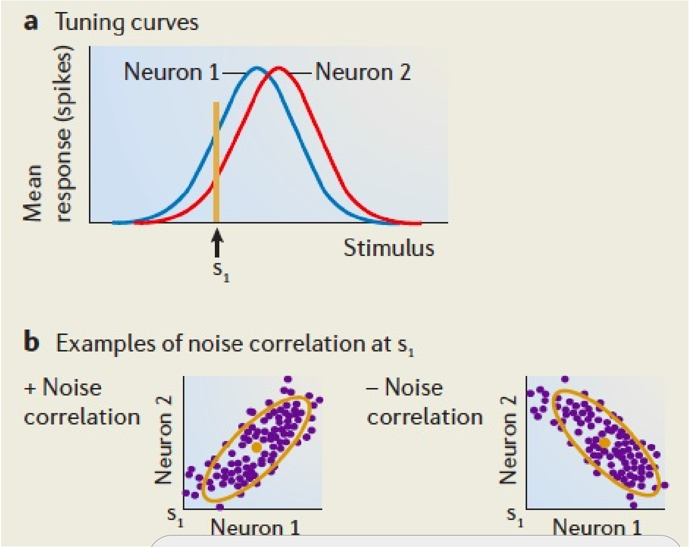
If plasticity depends on second-order (spiking) statistics, shouldn't the neuronal representations ("code") be based on them as well?

Traditional view: neuronal representations based on spike counts

Olfactory system



Visual system
tuning curves for
orientation selectivity



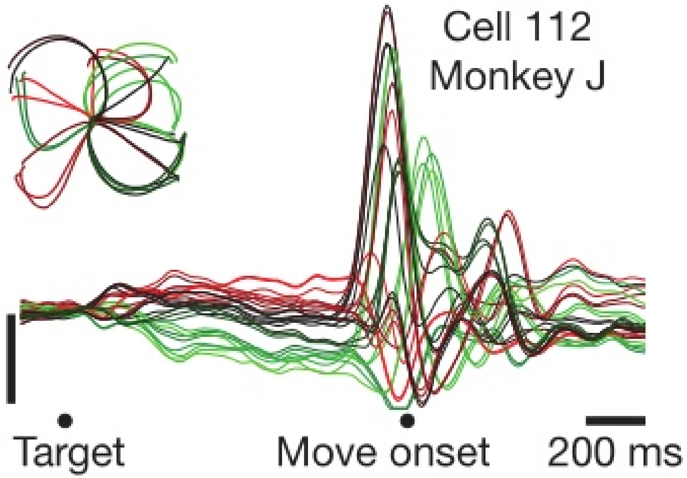
noise correlations
spike count across trials

Averbeck et al. (2006) *Nat Rev Neurosci*
Moreno-Bote et al. (2014) *Nat Neurosci*

Hige et al. *Nature* 2015

Temporal structure matters! Rate cofluctuations and spike synchrony

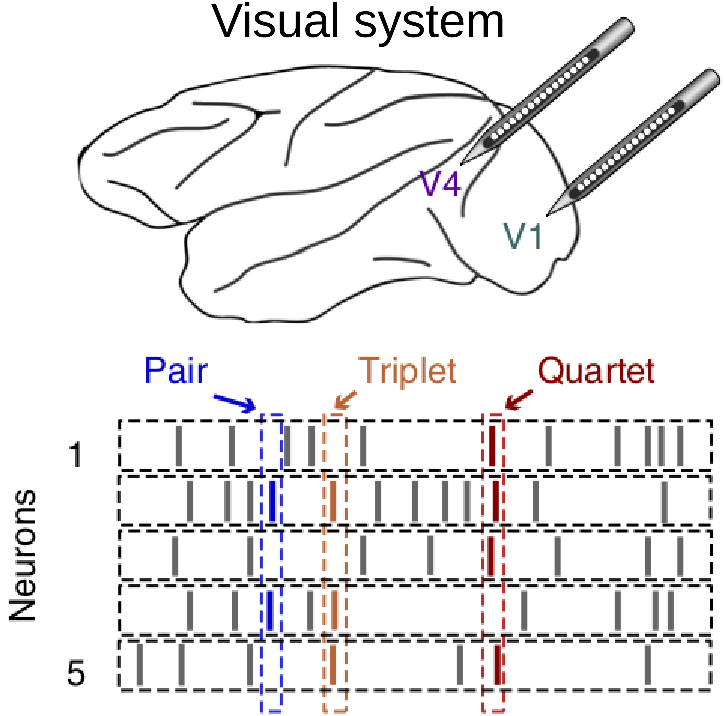
Motor system



fast rate correlation
within trial

Churchland et al. (2012) *Nature*
Vyas et al. (2020) *Ann Rev Neurosci*

Visual system



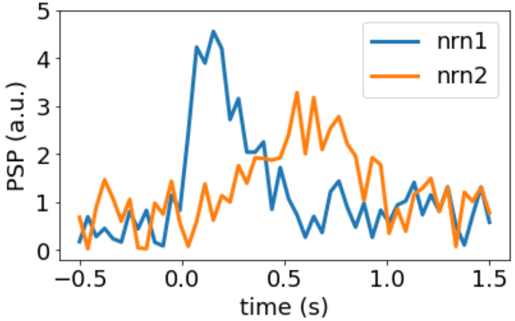
spike correlations
within trial

Shahidi et al (2019) *Nat Neurosci*

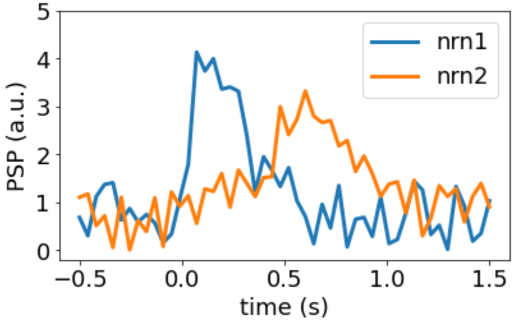
Types of structured variability in temporal (neuronal) signals: which measure to apply to time series?

Stable profile

Trial 1



Trial 2



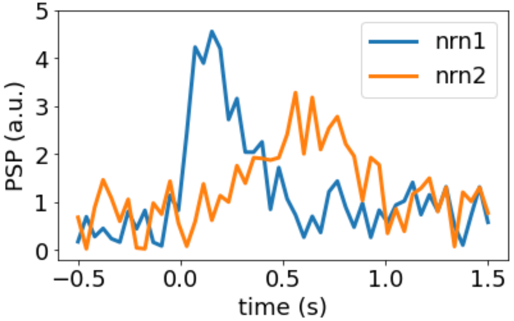
Types of structured variability in temporal (neuronal) signals: which measure to apply to time series?

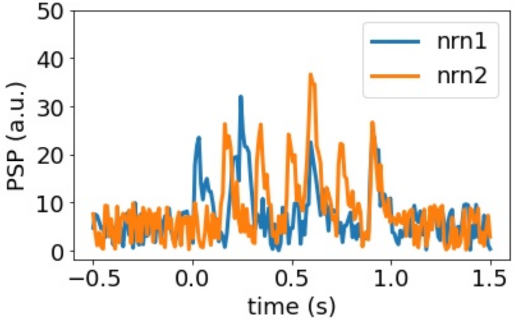
Stable profile

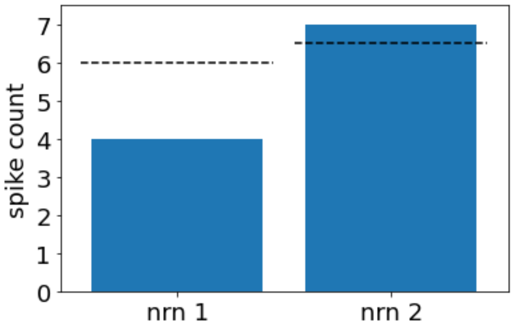
Coflucutations

Spike count

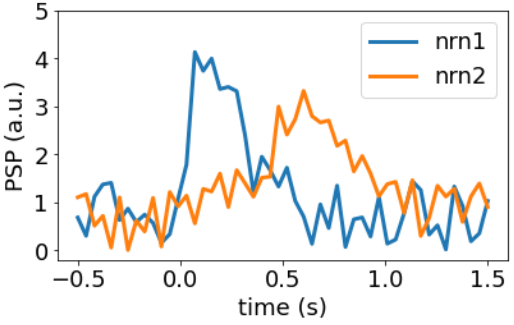
Trial 1

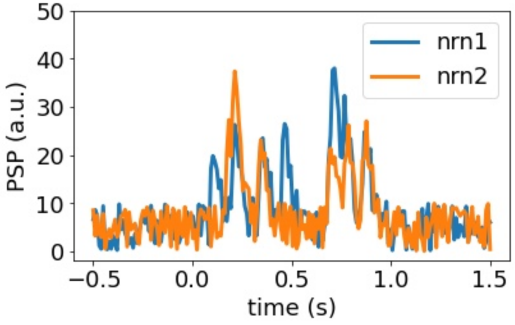


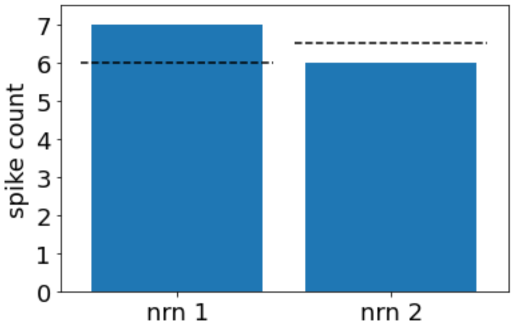




Trial 2







Types of structured variability in temporal (neuronal) signals: which measure to apply to time series?

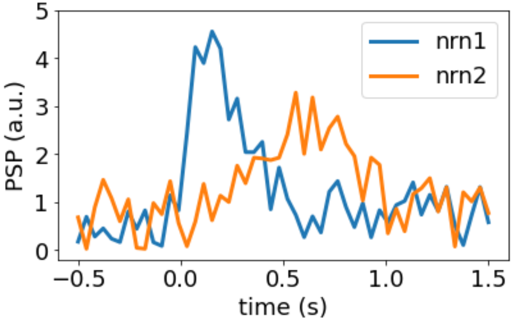
Stable profile

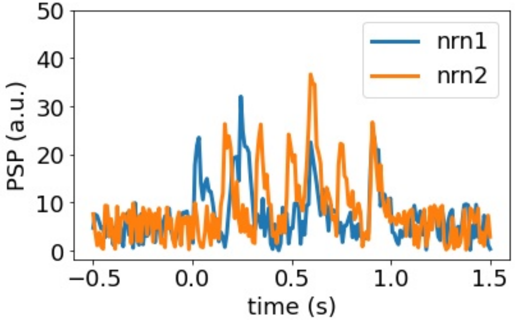
Coflutuations

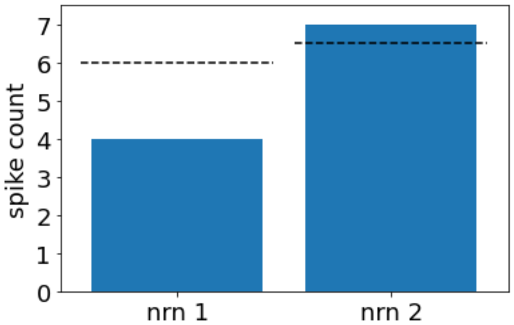
Second-order statistics
within trial correlations

Spike count

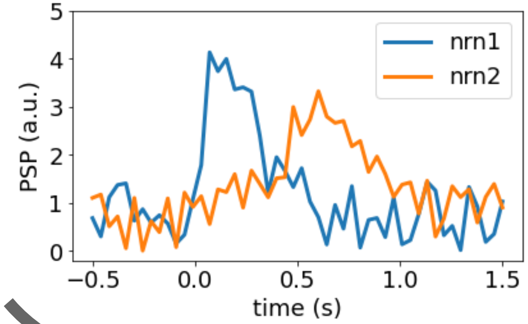
Trial 1

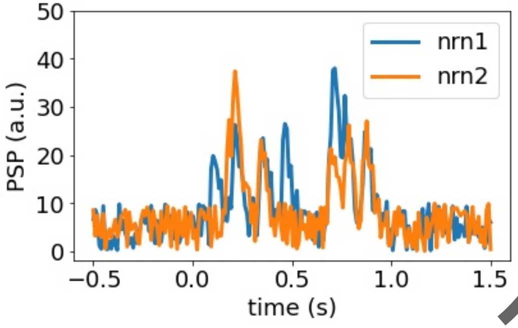


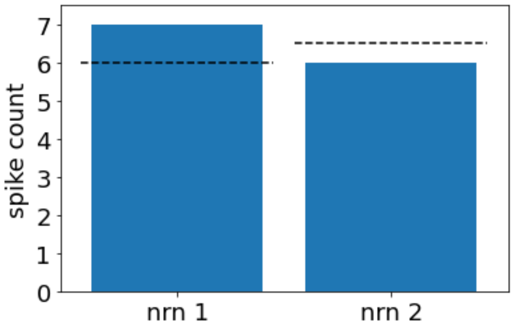




Trial 2

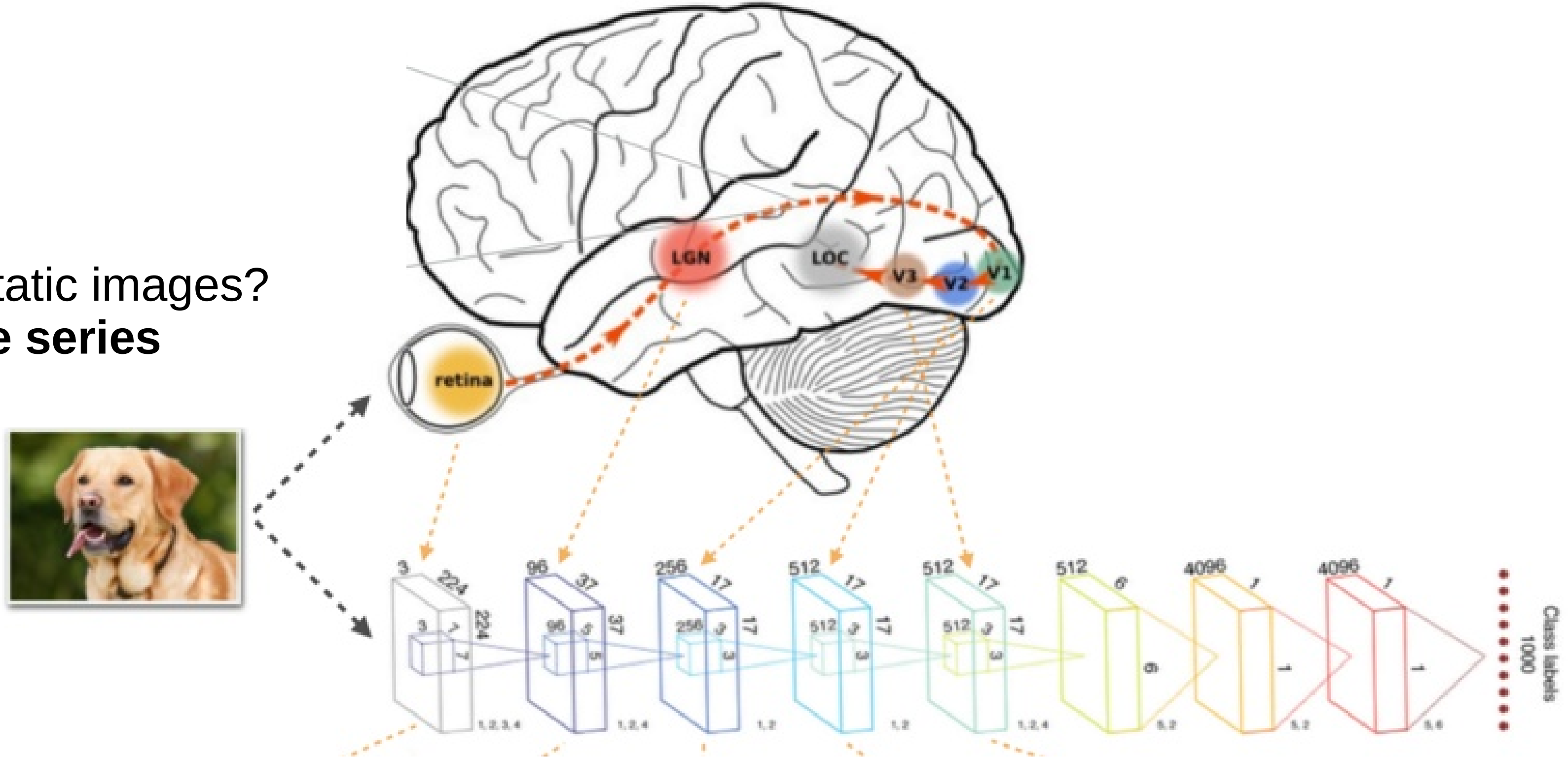






Computations: biological processing versus machine learning

Beyond static images?
Time series



Deep learning networks: what is processed at each layer?
Interpretability

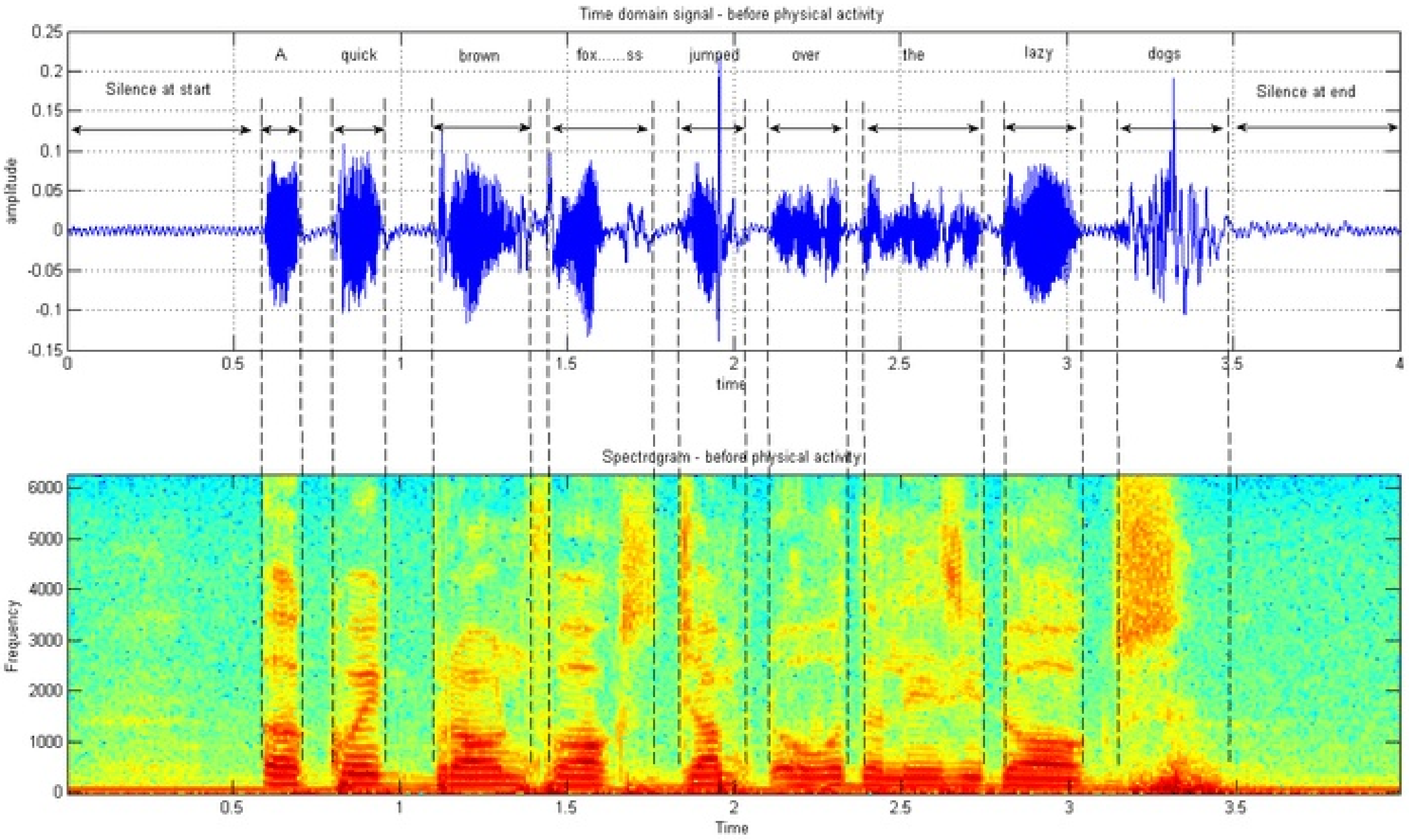
Outline

- **Statistical learning for time series**
  - **mean versus covariance decoding**
  - processing by neuronal reservoir
  - decoding by biological architecture
- Theory

Phonemes in speech

soundwave

spectrogram

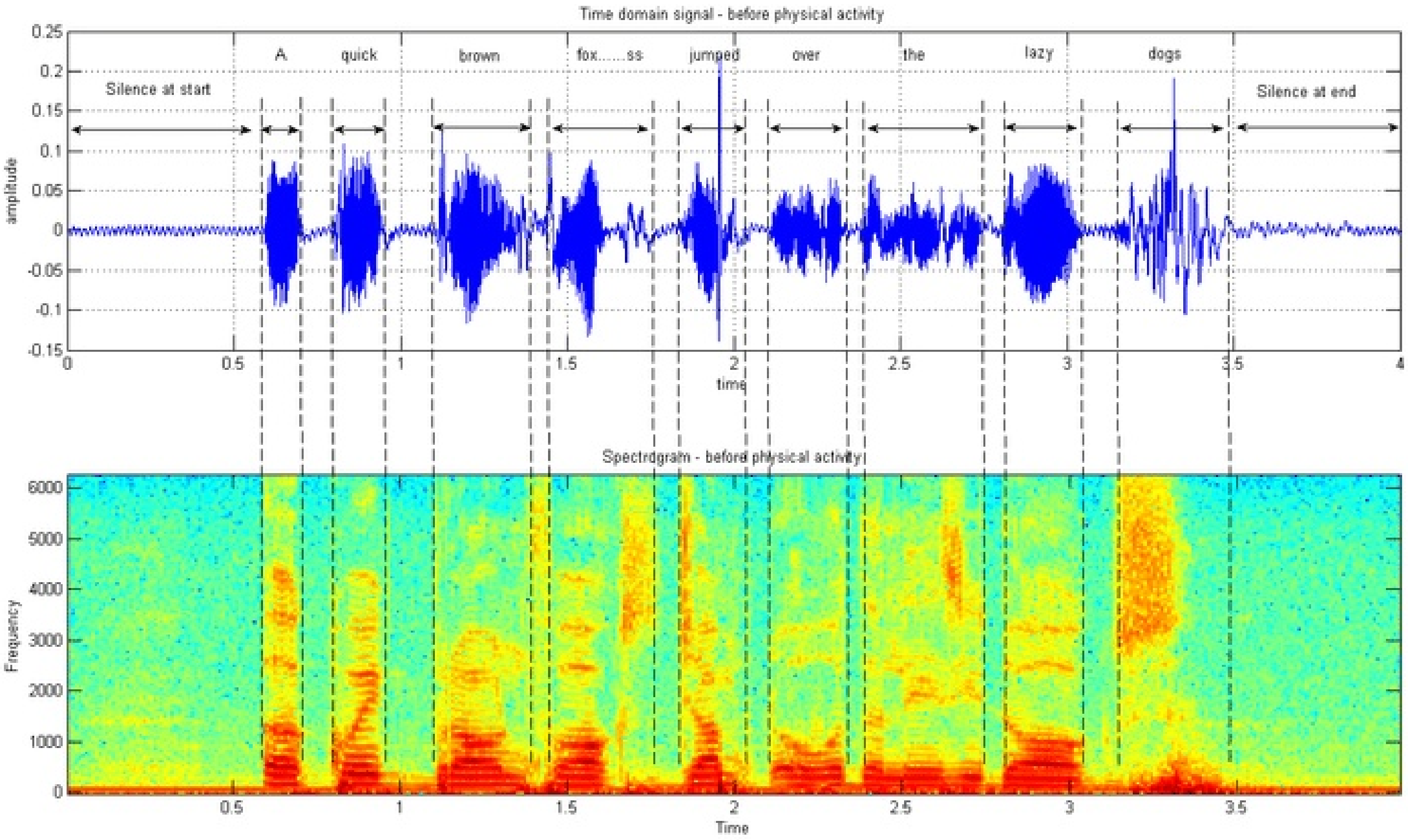


http://timeseriesclassification.com

Phonemes in speech

soundwave

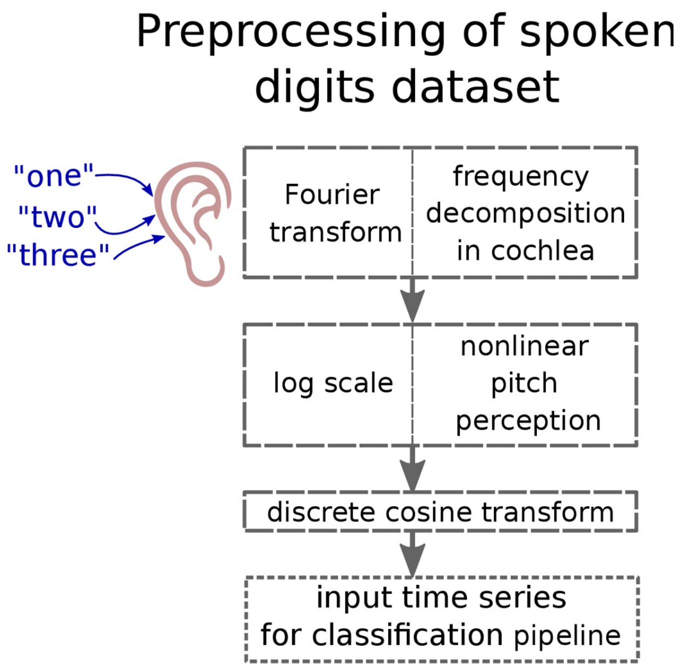
spectrogram

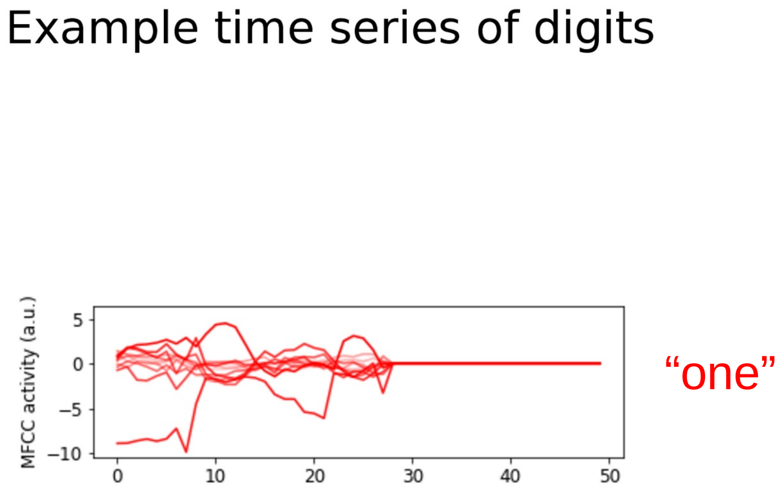


inner ear: cochlea

http://timeseriesclassification.com

Cochlear processing and spatio-temporal structure

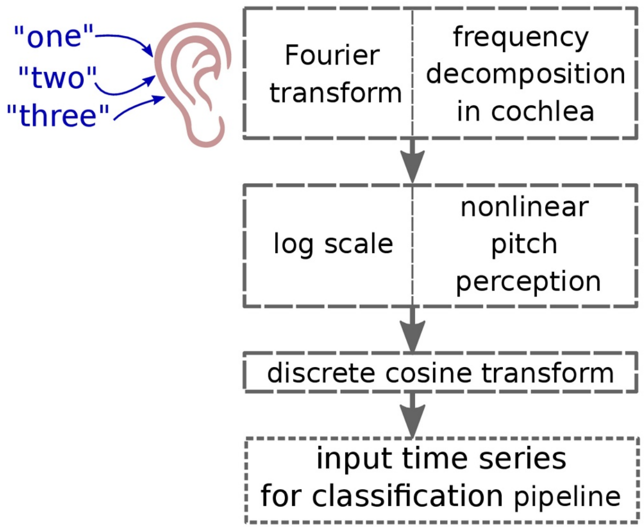




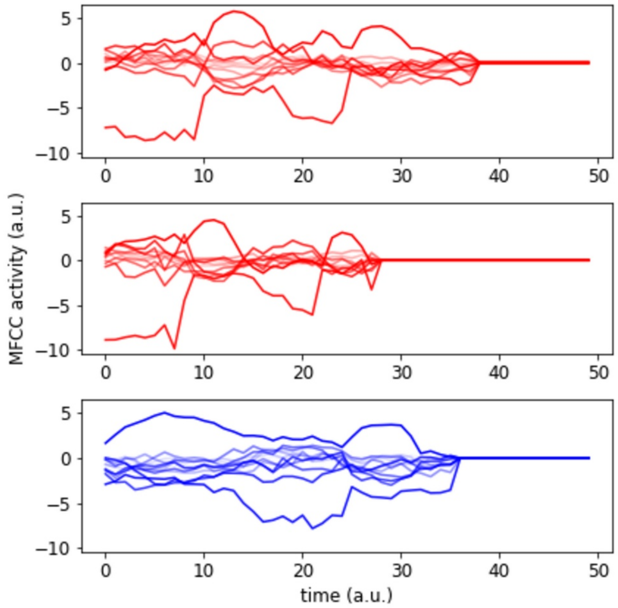
http://timeseriesclassification.com

Cochlear processing and spatio-temporal structure

Preprocessing of spoken digits dataset



Example time series of digits



“oone”

“one”

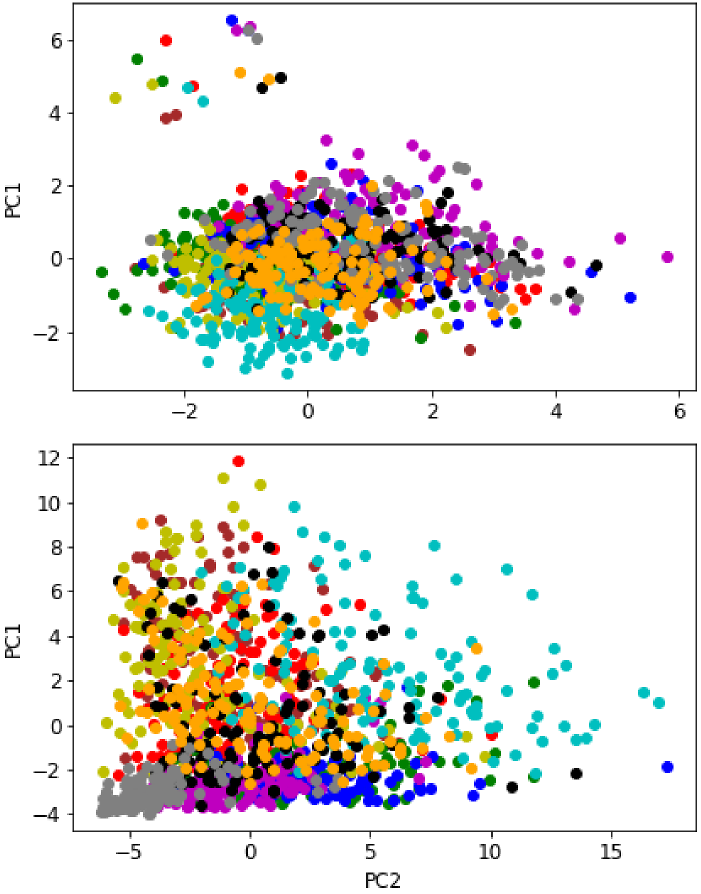
“three”

Separability of spoken digits (10 classes)

PCA

mean

cov



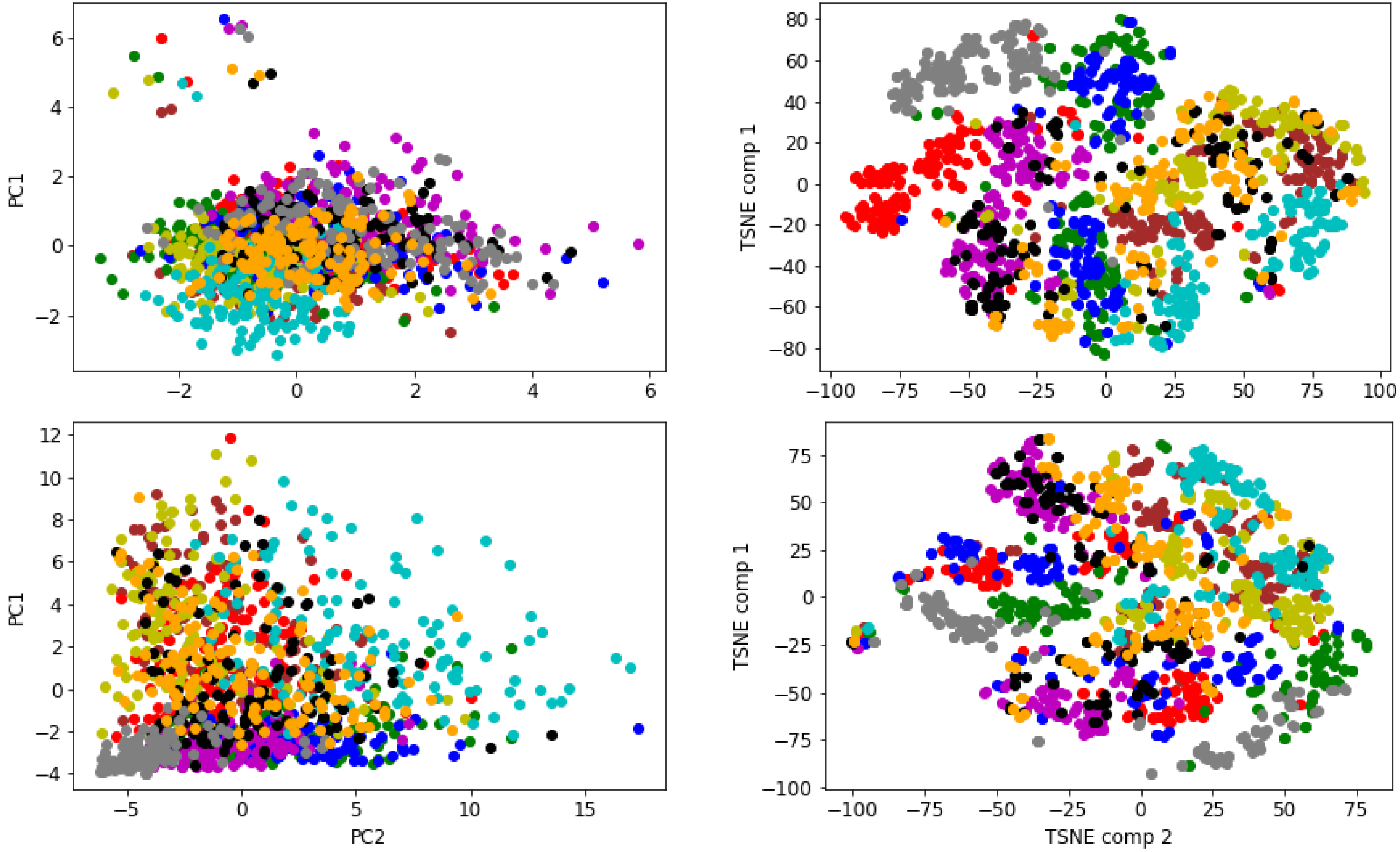
Separability of spoken digits (10 classes)

PCA

tSNE

mean

cov



Cross-validated classification to assess baseline separability

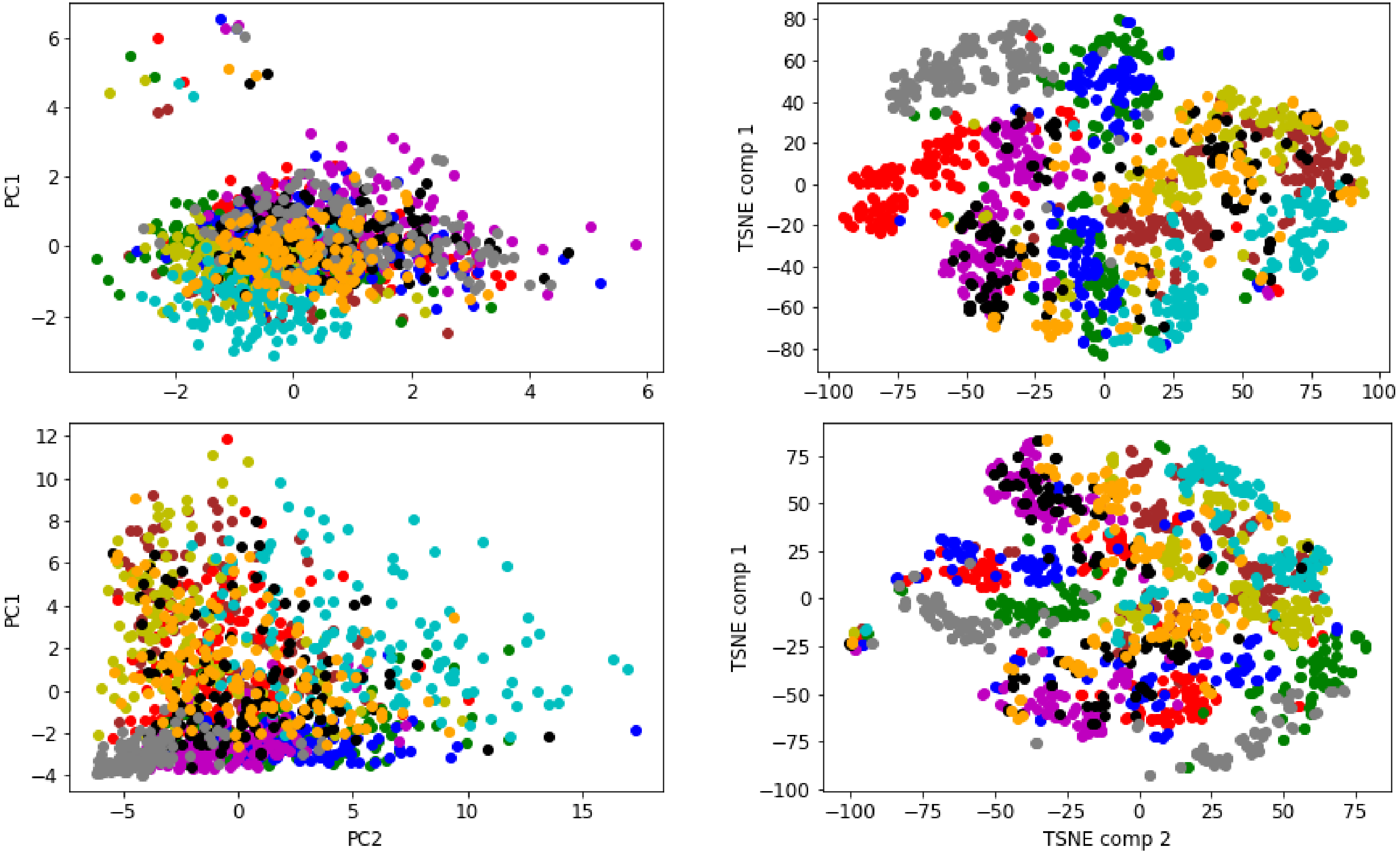
PCA

tSNE

mean **69.7%**

Logistic regression

cov **91.7%**



Outline

- **Statistical learning for time series**
  - mean versus covariance decoding
  - processing by neuronal reservoir
  - decoding by biological architecture
- Theory

- **Structured variability conveys information**

Outline

- **Statistical learning for time series**
  - mean versus covariance decoding
  - **processing by neuronal reservoir**
  - decoding by biological architecture
- Theory

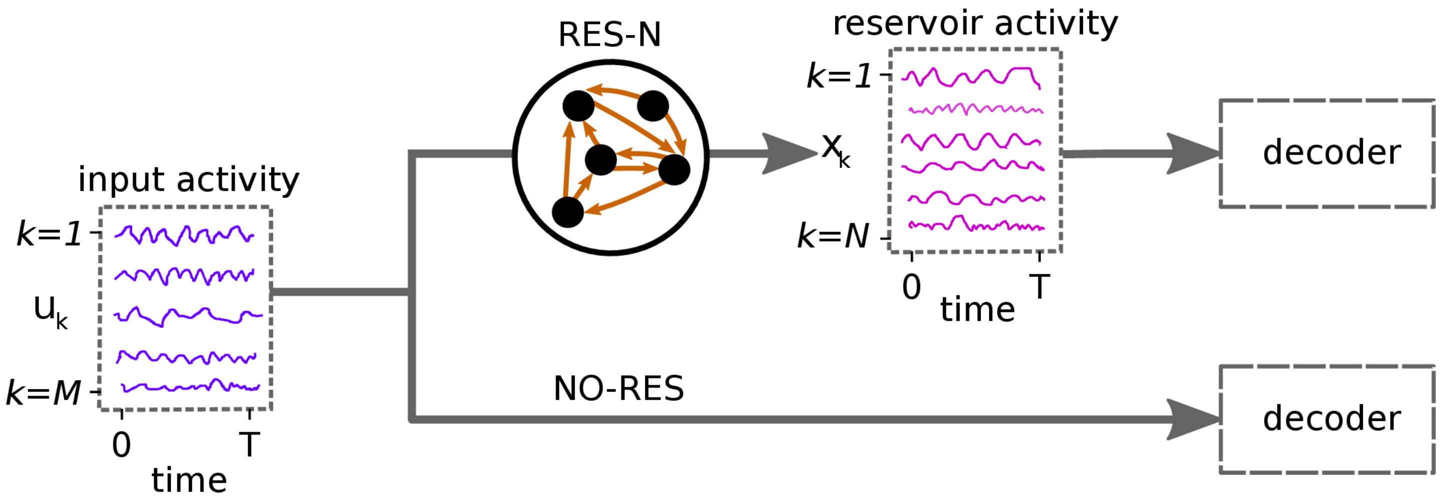
- Structured variability conveys information
- **How to efficiently extract statistical patterns?**

Reservoir computing to process input time series

Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

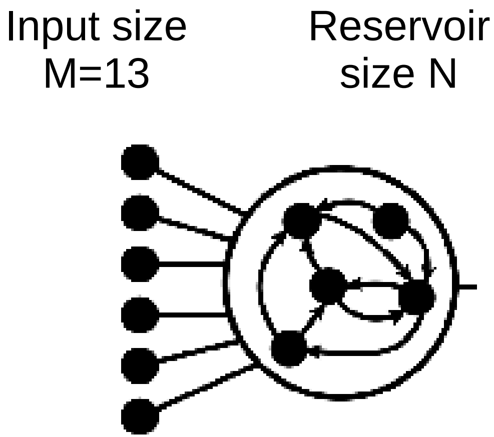
Reservoir computing to process input time series

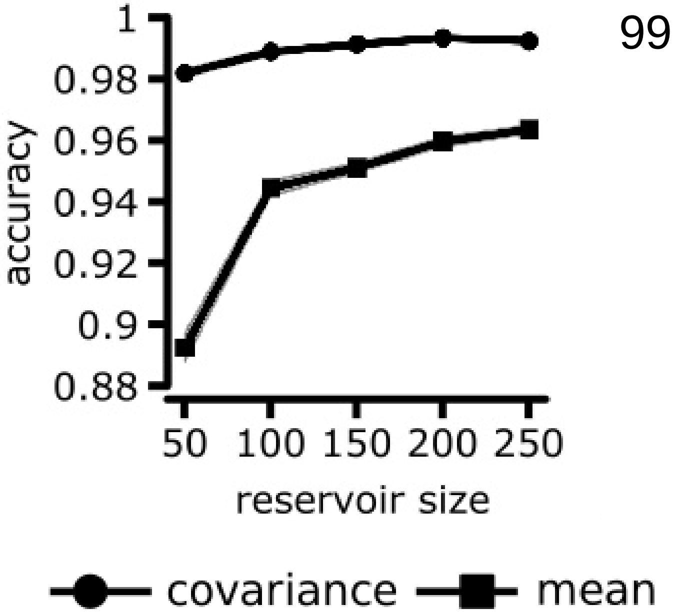
Echo-state network
(Jaeger style)



Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Dimensionality expansion



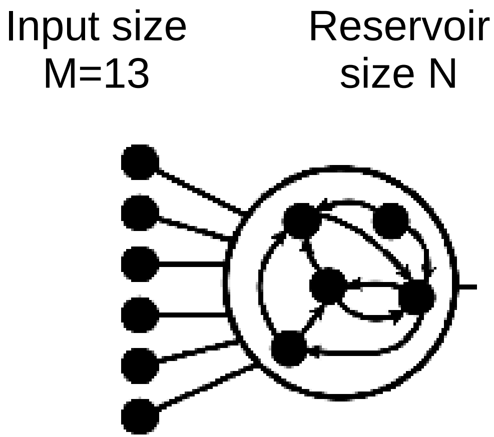


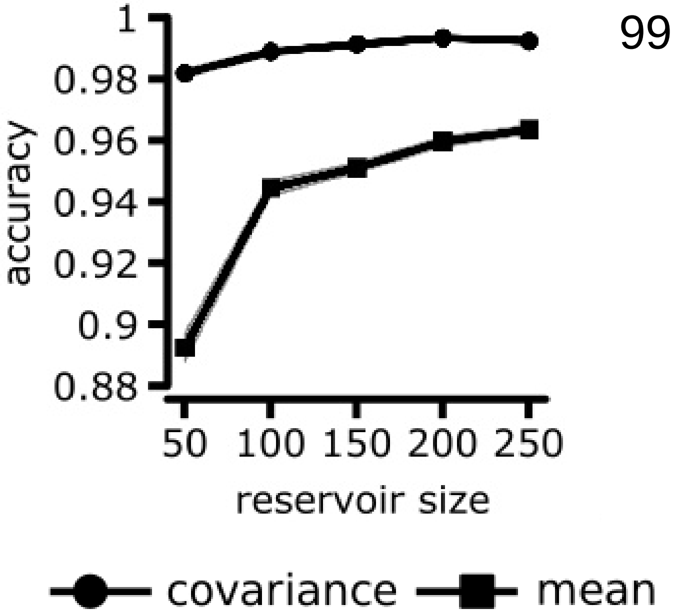
99.2%

NO-RES ref: 91.7% 69.7%

Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Dimensionality expansion





99.2%

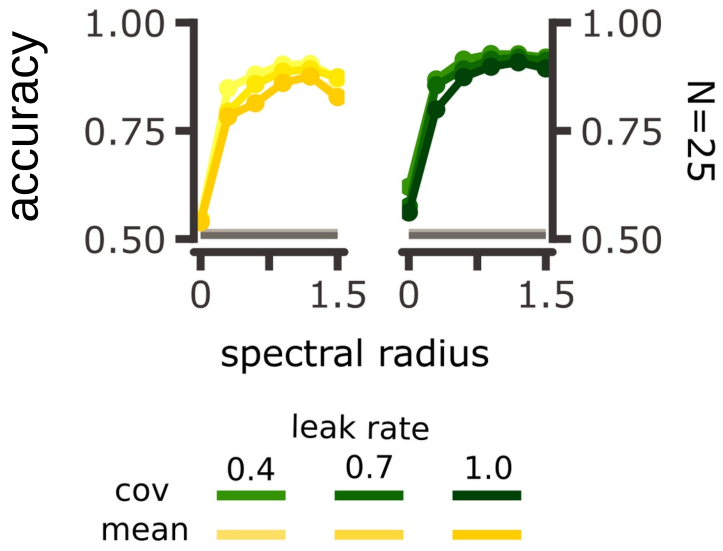
Reservoir feature space:

- mean vector **N**
- cov matrix **N²**

NO-RES ref: 91.7% 69.7%

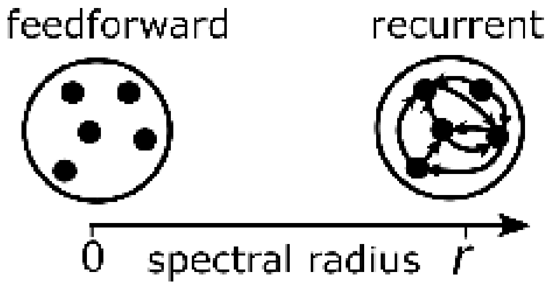
Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Influence of reservoir parameters on decoding accuracy?



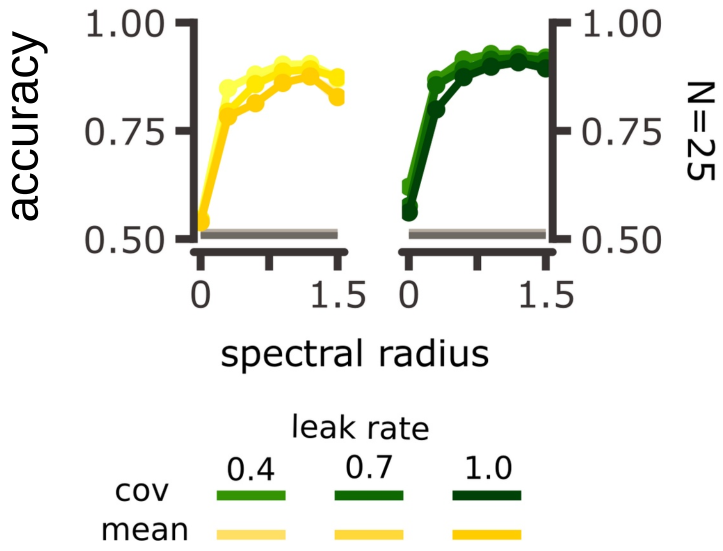
Parameter exploration:

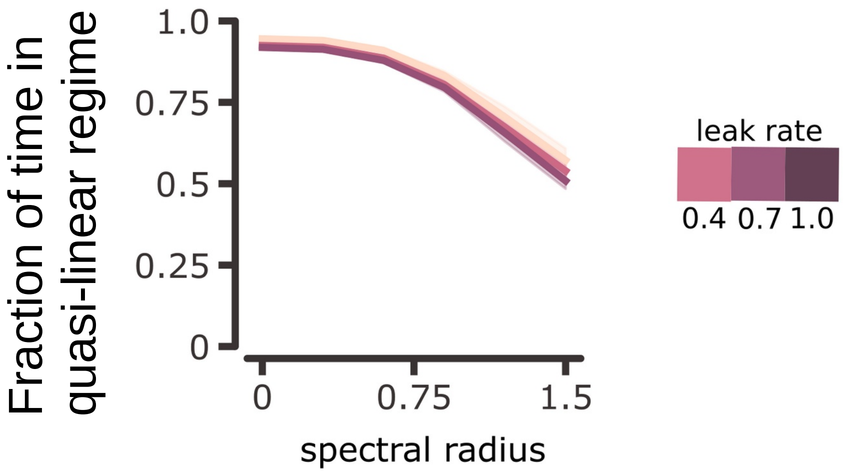
- spectral radius
- nodal leak rate



Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

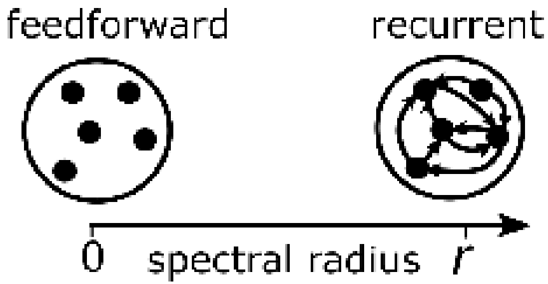
Influence of reservoir parameters on decoding accuracy? Nonlinear reservoir dynamics support best decoding





Parameter exploration:

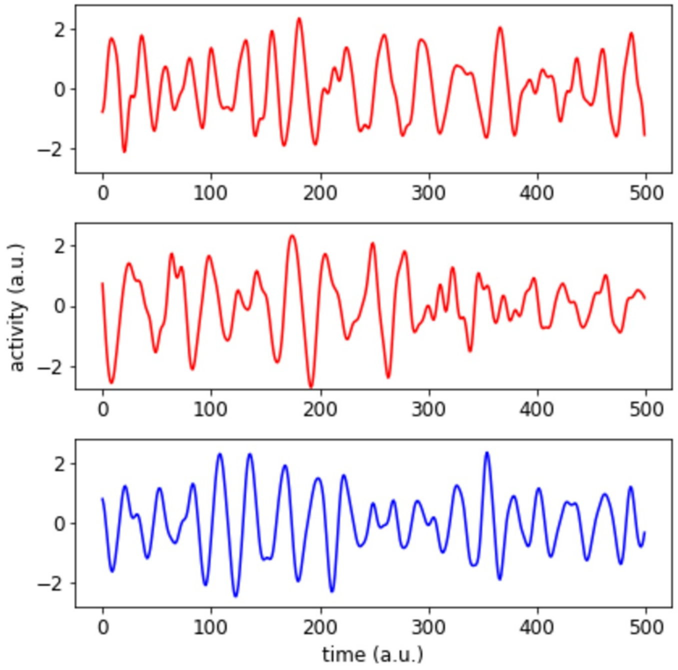
- spectral radius
- nodal leak rate



Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Example dataset 2: faulty engine signals

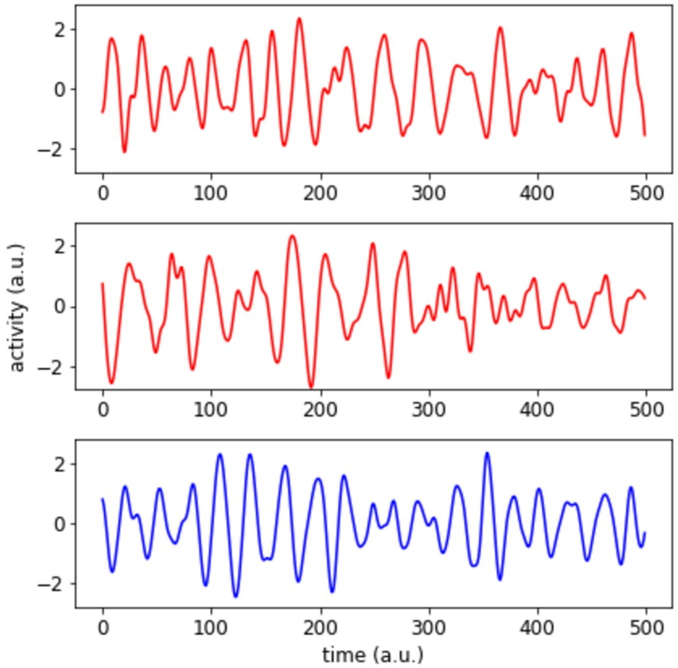
Example time series for
faulty engine signals



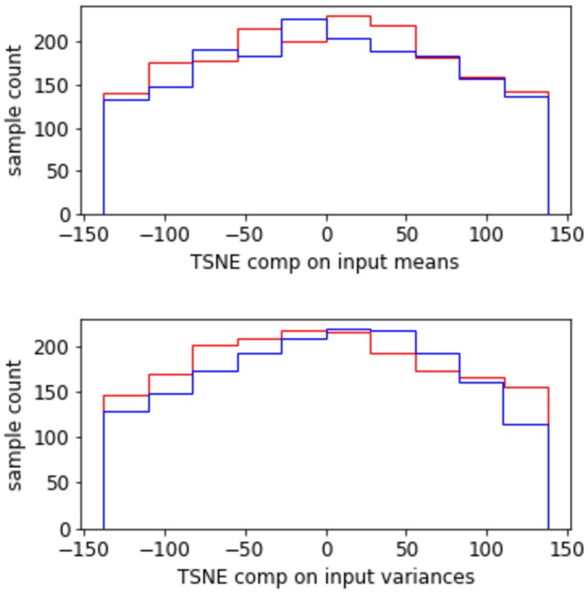
Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Example dataset 2: faulty engine signals

Example time series for faulty engine signals



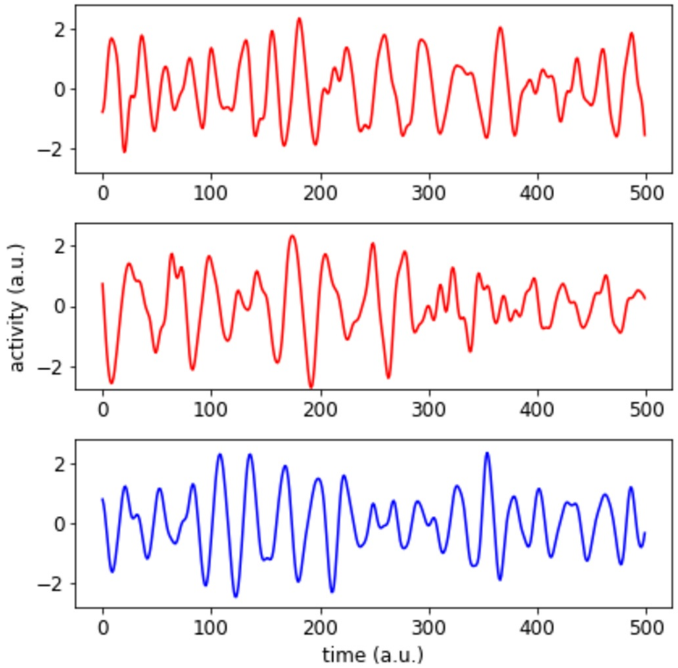
TSNE on means/variances for faulty engine signals



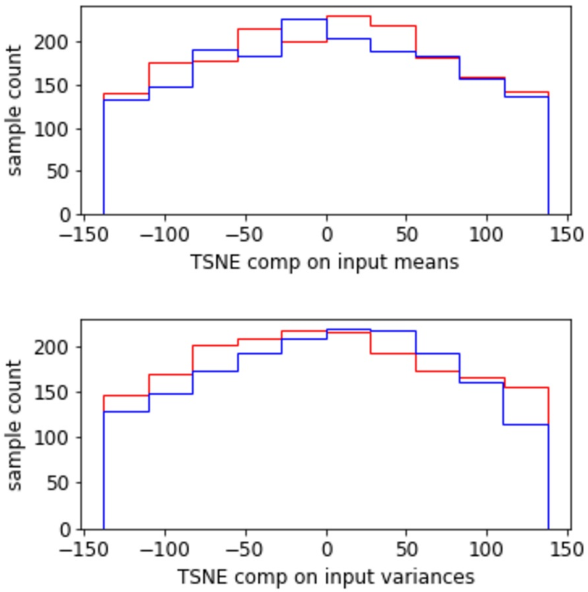
Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Example dataset 2: faulty engine signals

Example time series for faulty engine signals



TSNE on means/variances for faulty engine signals



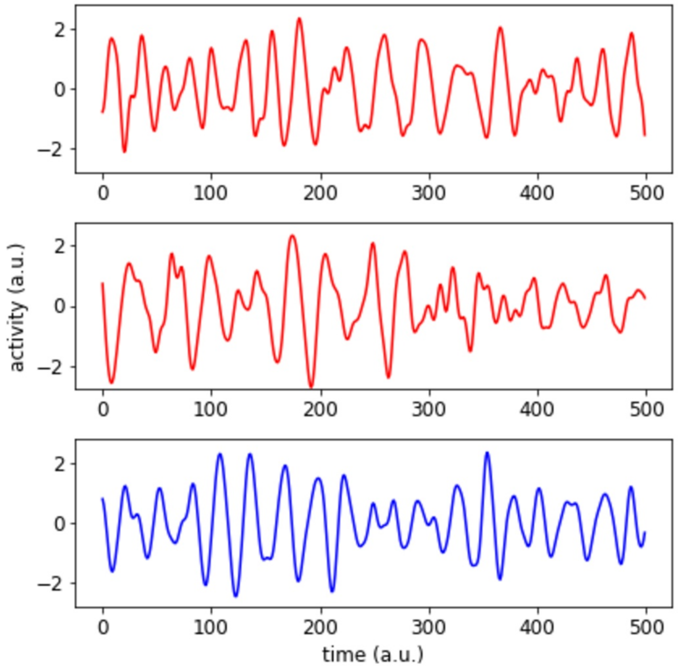
NO-RES

- **mean: ~50%**
- **cov: ~50%**
- **best: 96%**

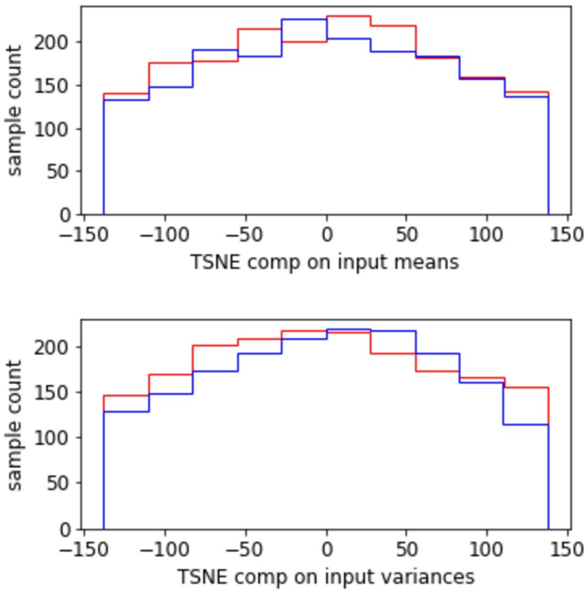
Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

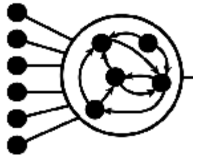
Example dataset 2: faulty engine signals

Example time series for faulty engine signals



TSNE on means/variances for faulty engine signals





(size N=50)

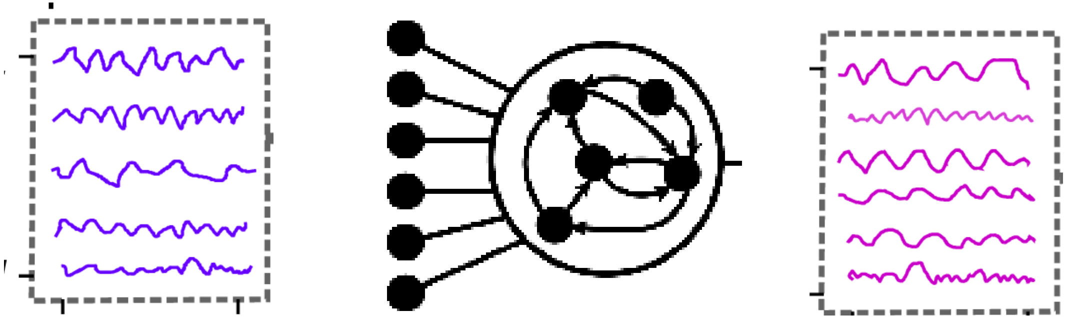
- **mean: 84%**
- **cov: 94%**
- **best: 96%**

Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Neuronal reservoir for time series processing

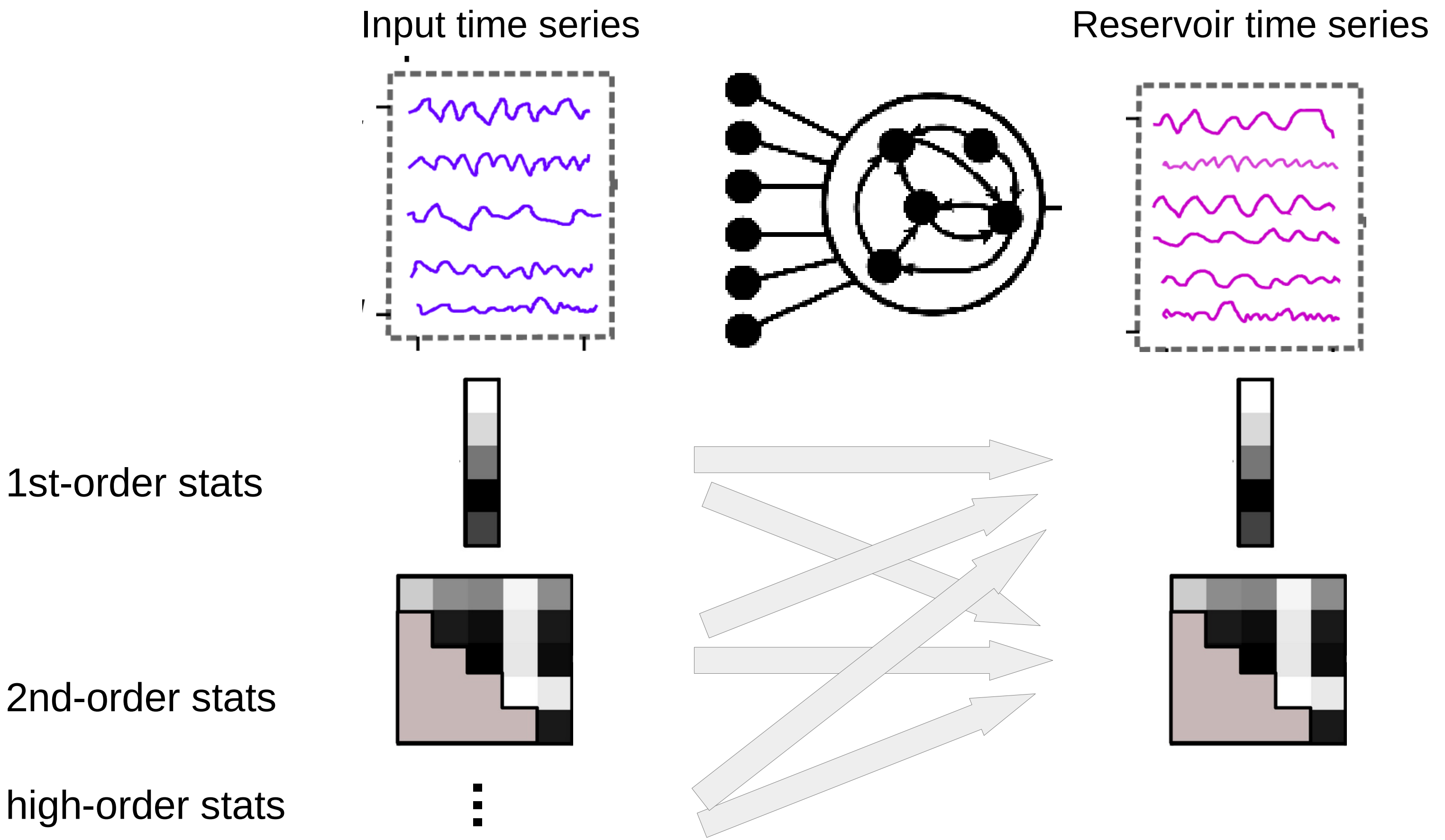
Input time series

Reservoir time series



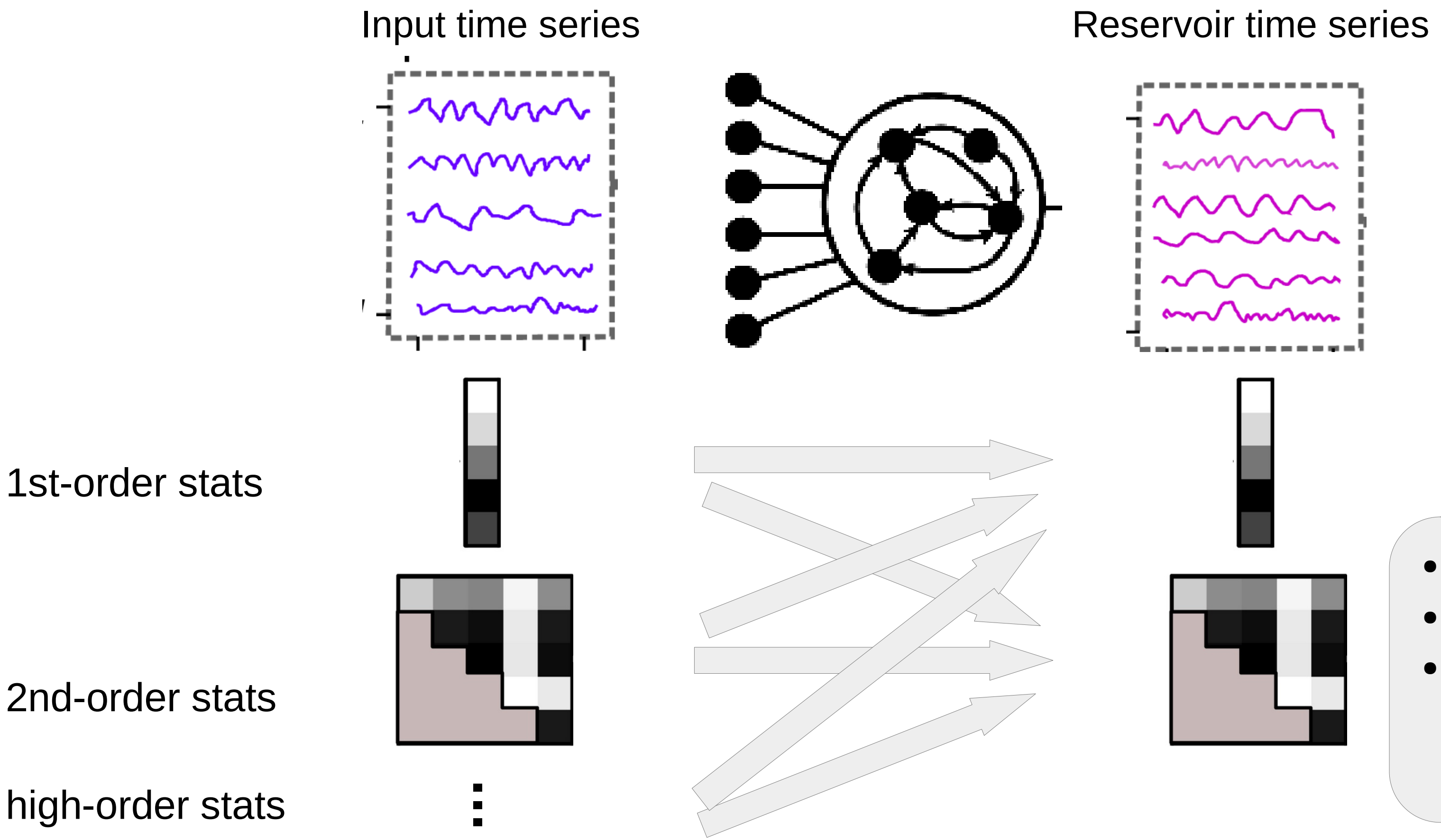
Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Neuronal reservoir for time series processing



Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Neuronal reservoir for time series processing



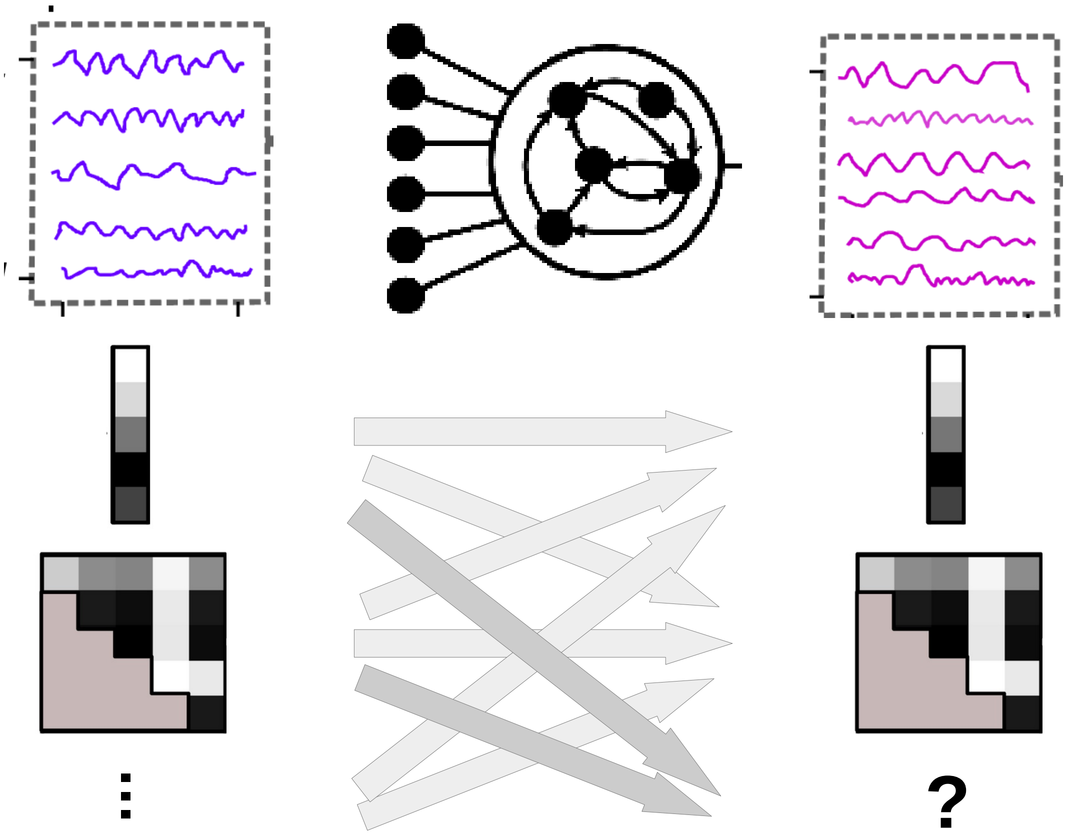
- recurrent connectivity
- nodal nonlinearity
- need to improve theory away from linear regime

Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Neuronal reservoir for time series processing

Input time series

Reservoir time series



1st-order stats

2nd-order stats

high-order stats

- recurrent connectivity
- nodal nonlinearity
- need to improve theory away from linear regime

Lawrie, Moreno-Bote, Gilson *ICCVBIC* 2021; *biorxiv* (in revision)

Outline

- **Statistical learning for time series**
  - mean versus covariance decoding
  - processing by neuronal reservoir
  - decoding by biological architecture
- Theory

- Structured variability conveys information

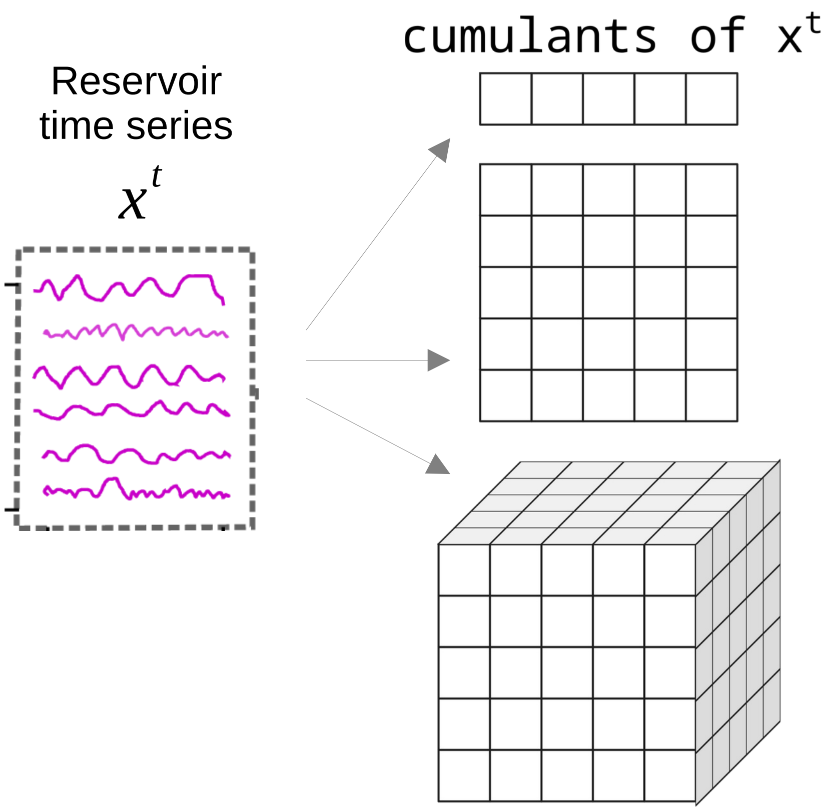
- **Recurrent connectivity + nonlinearity are key**

Outline

- **Statistical learning for time series**
  - mean versus covariance decoding
  - processing by neuronal reservoir
  - **decoding by biological architecture**
- Theory

- Structured variability conveys information
- Recurrent connectivity + nonlinearity are key
- **Extension to high-order statistics?**

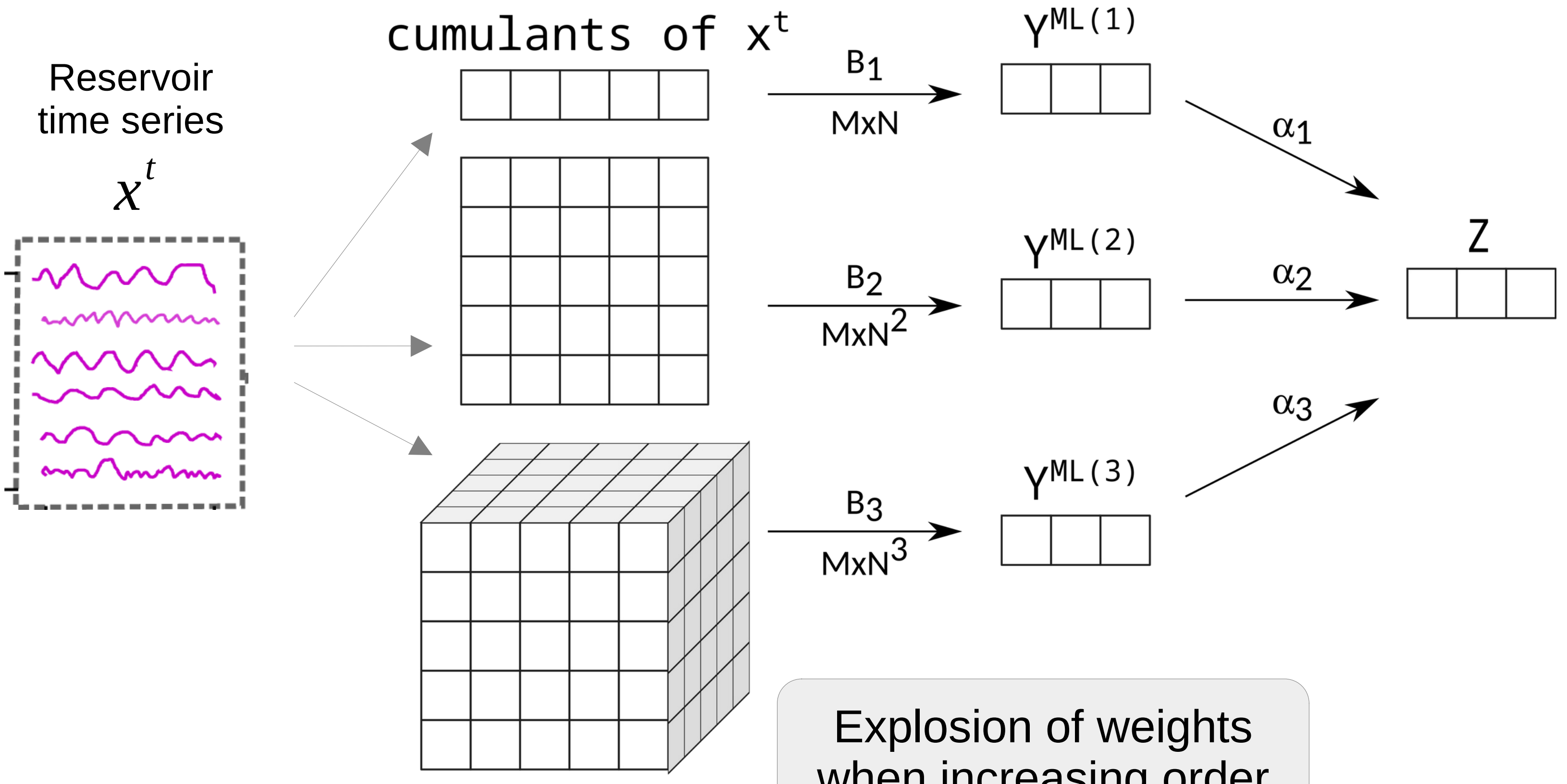
Extension to statistical orders 1 to 3



3 classes
to predict

Nestler, Helias, Gilson (submitted) *arxiv*

Extension to statistical orders 1 to 3

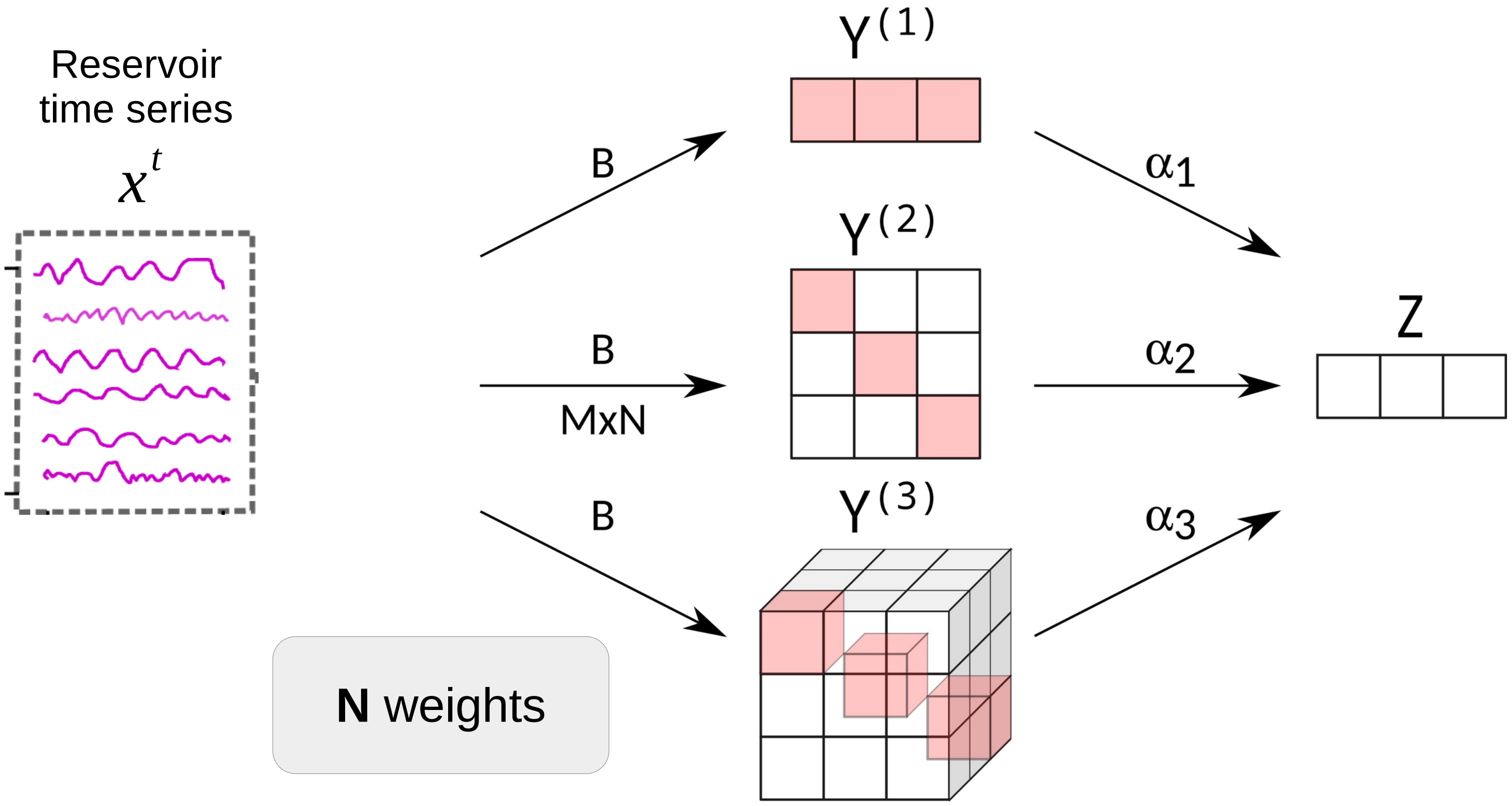


3 classes to predict

Explosion of weights when increasing order

Nestler, Helias, Gilson (submitted) *arxiv*

Extension to statistical orders 1 to 3



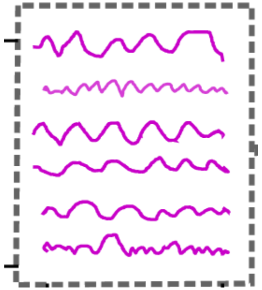
N weights

3 classes
to predict

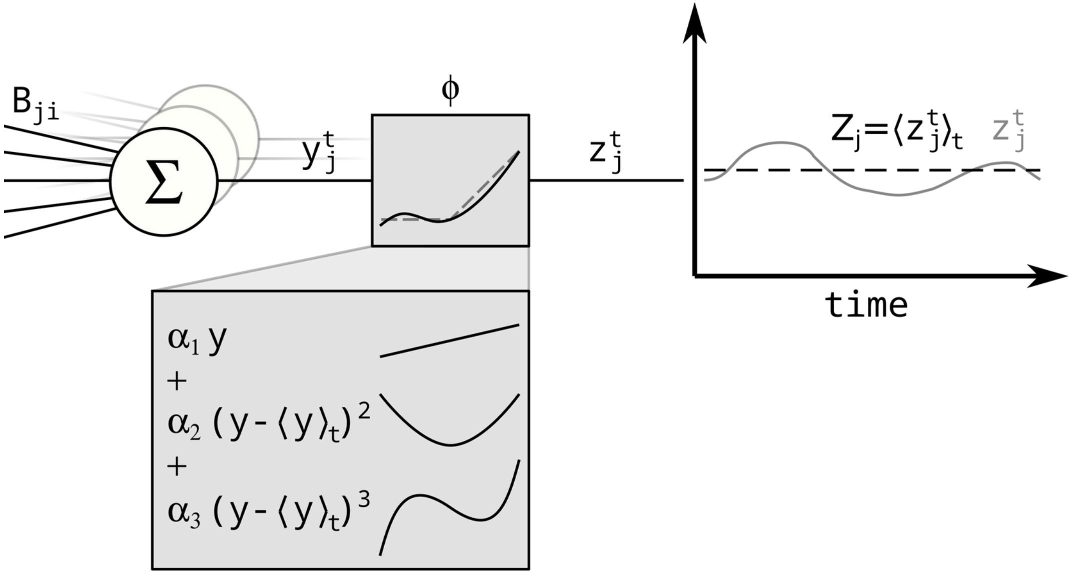
Nestler, Helias, Gilson (submitted) *arxiv*

Order-selective perceptron (OSP)

Reservoir
time series
x^t



Tunable nonlinearity:
gradient descent
(+ back-prop for weights)



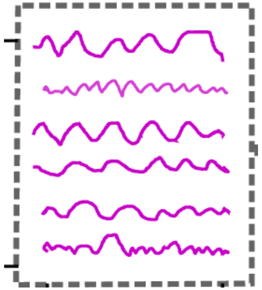
Decision
based on
mean
output

Nestler, Helias, Gilson (submitted) *arxiv*

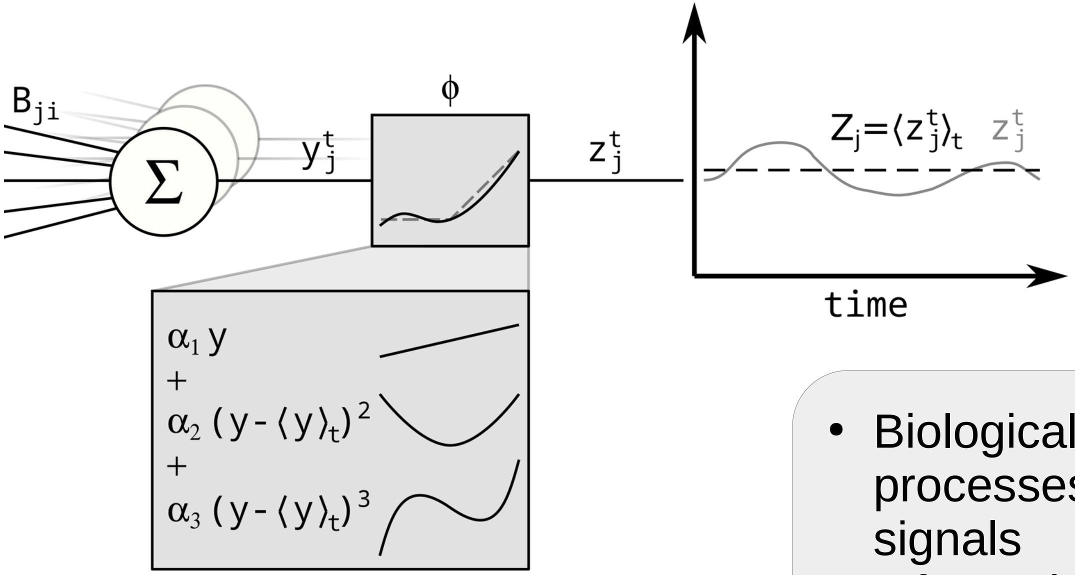
Order-selective perceptron (OSP)

Reservoir
time series

x^t



Tunable nonlinearity:
gradient descent
(+ back-prop for weights)

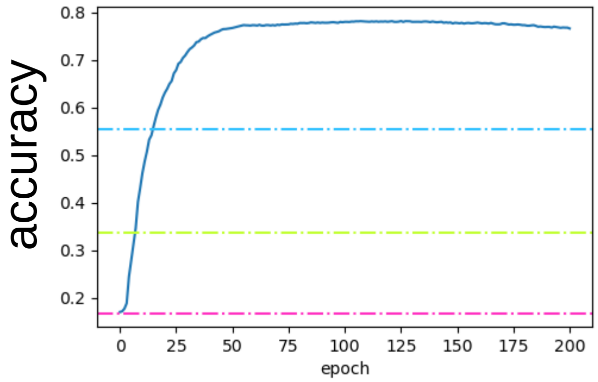


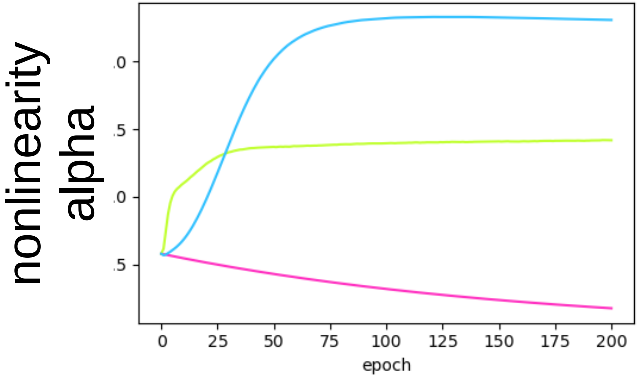
Decision
based on
mean
output

- Biological system that processes temporal signals
- Information embedded in time series

Nestler, Helias, Gilson (submitted) *arxiv*

Interpretable learning





3rd order only

2nd order only

1st order only

Several orders are effectively used

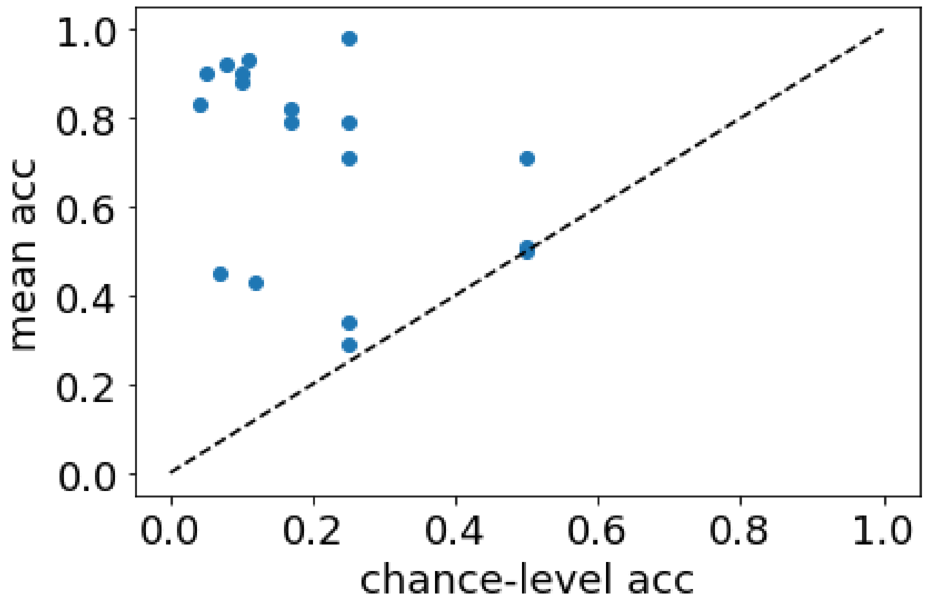
Which order is relevant?

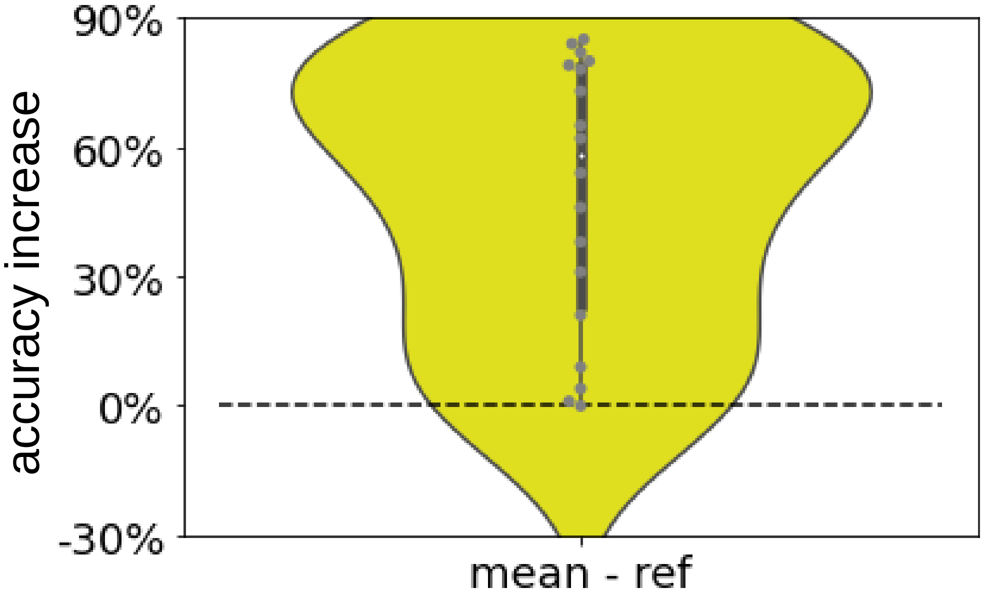
Nestler, Helias, Gilson (submitted) *arxiv*

Comparison on real multivariate datasets

Hand movements (video), epilepsy (EEG), sensors, etc.

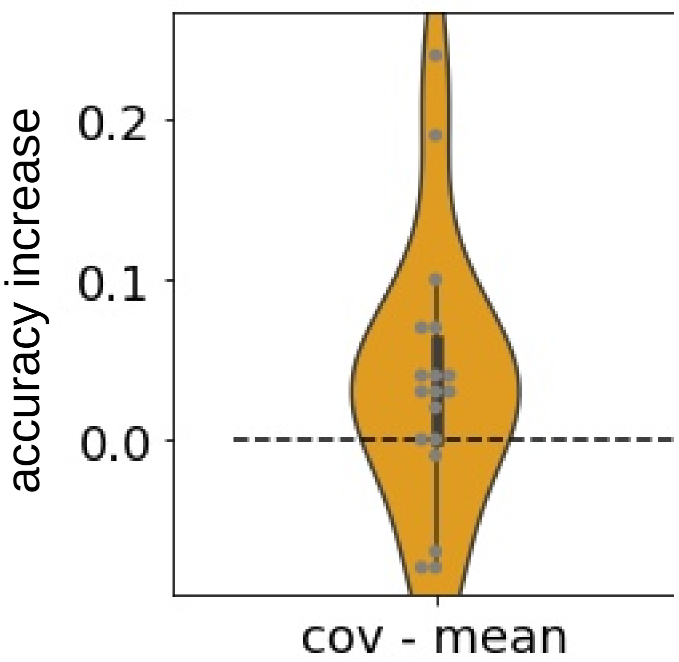
=100%





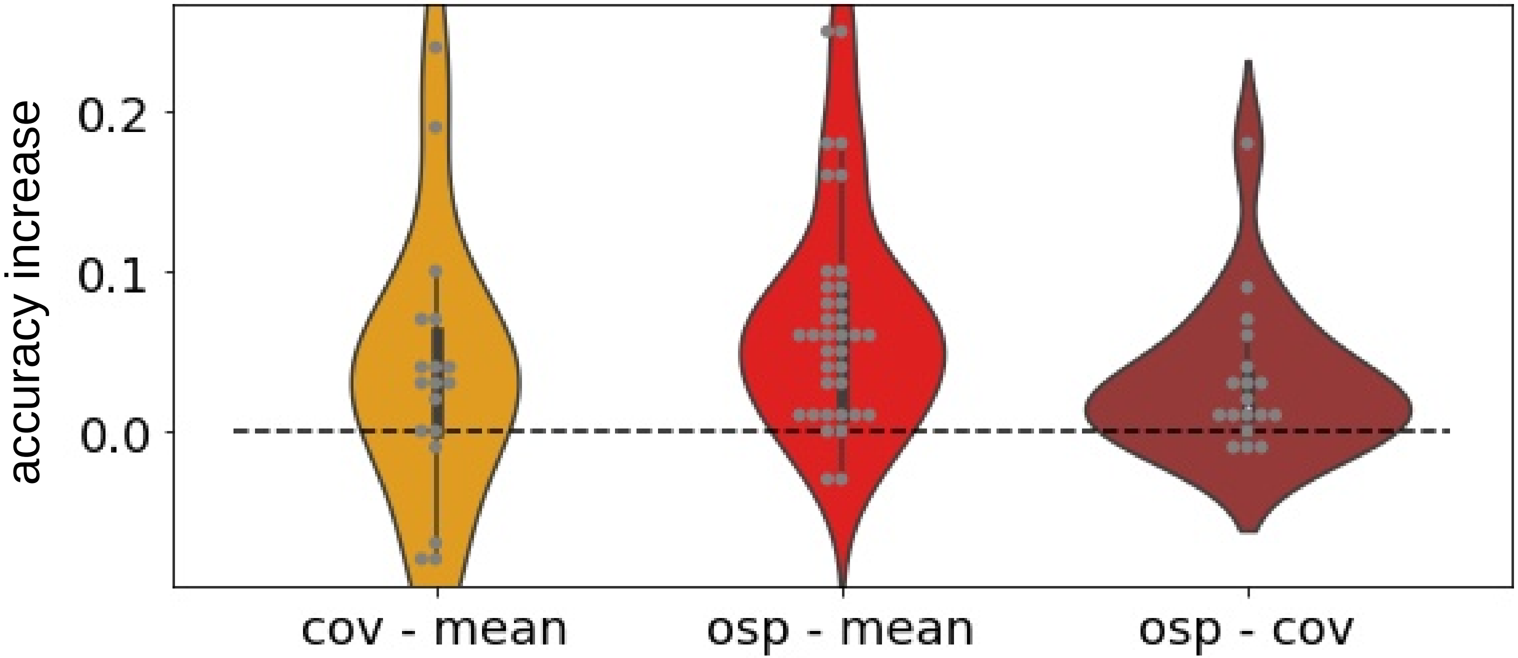
http://timeseriesclassification.com

Comparison on real multivariate datasets (reservoir + decoding)



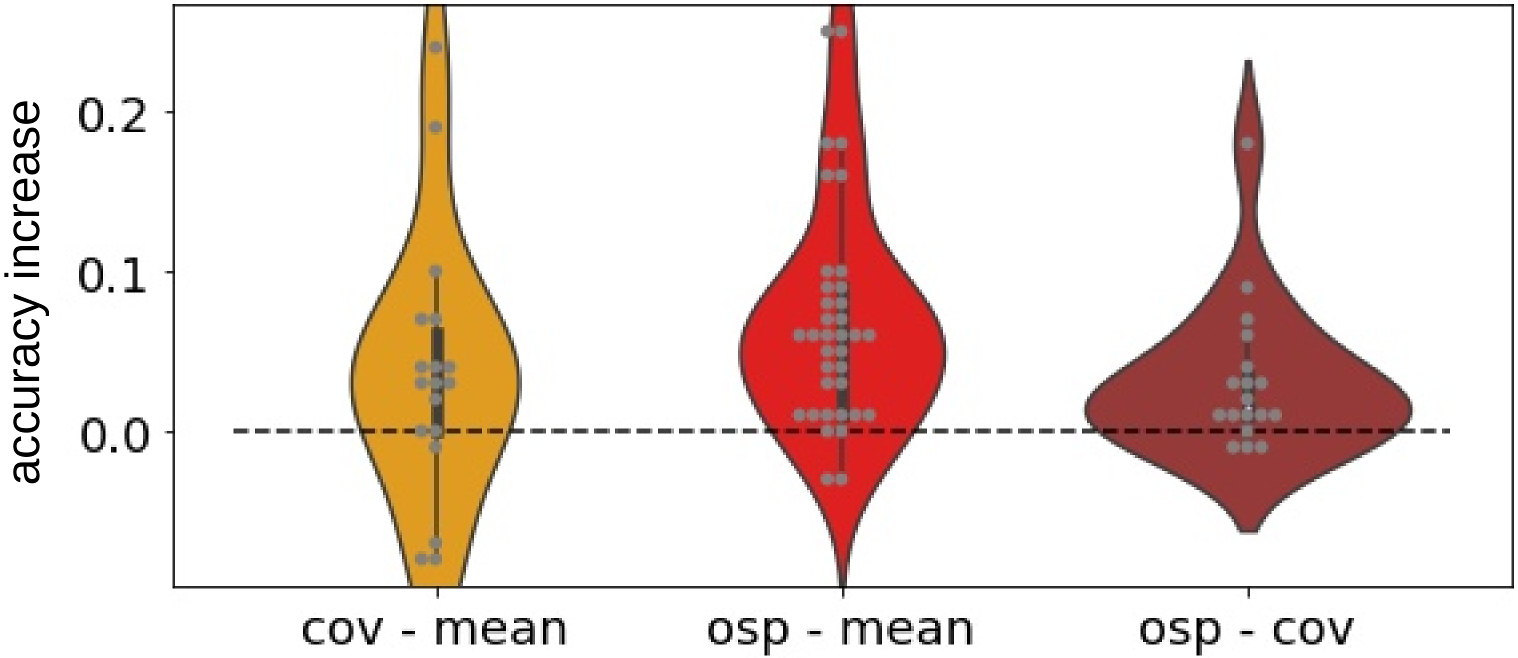
Covariance decoding
outperforms
mean decoding
in most cases

Comparison on real multivariate datasets (reservoir + decoding)



Synergy? Yes!

Comparison on real multivariate datasets (reservoir + decoding)



Synergy? Yes!

N versus **N²** weights!

http://timeseriesclassification.com

Outline

- **Statistical learning for time series**
  - mean versus covariance decoding
  - processing by neuronal reservoir
  - decoding by biological architecture
- Theory

- Structured variability conveys information

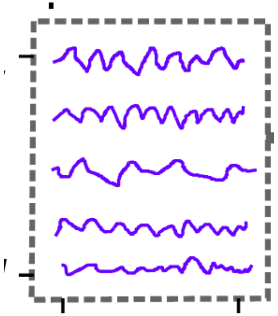
- Recurrent connectivity + nonlinearity are key

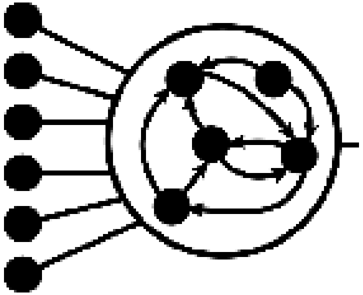
- **Robust decoding with limited resources**

Next steps:
- **Trainable reservoir**
- **Deep net architecture**

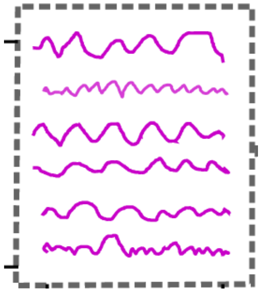
Conclusion on biological system

Input time series

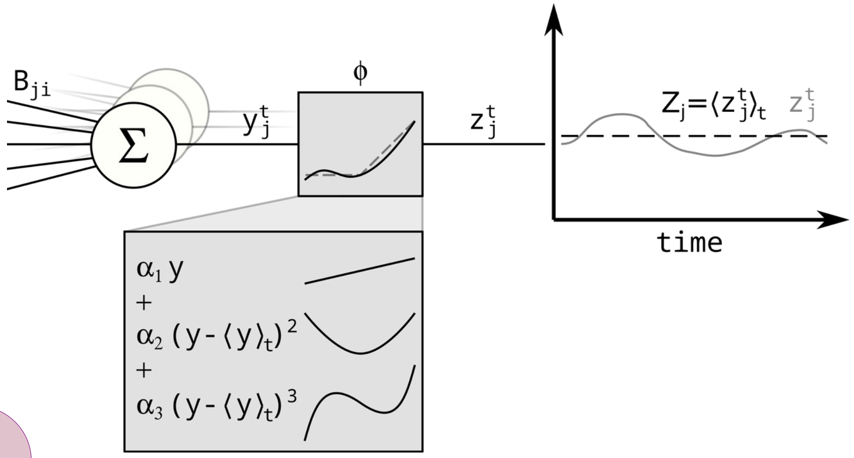




Reservoir time series



Readout



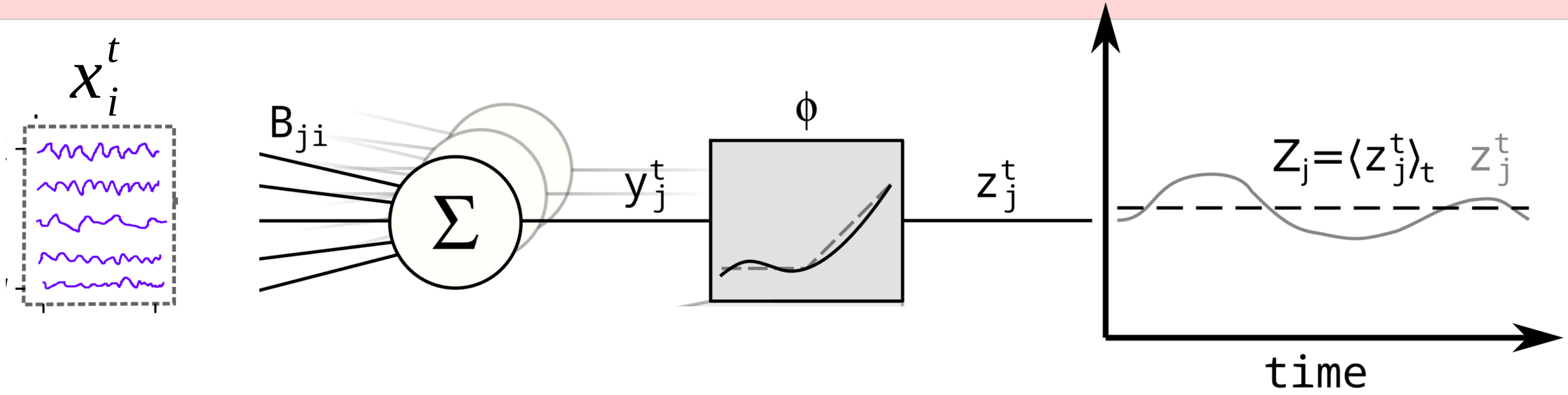
- Rich representation
- Expansion and cross-talk across statistical orders
- Recurrent connectivity + nonlinearity

- Automated selection of relevant statistical orders
- Mapping to first-order in output

Outline

- Statistical learning for time series
  - mean versus covariance decoding
  - processing by neuronal reservoir
  - decoding by biological architecture
- **Theory**

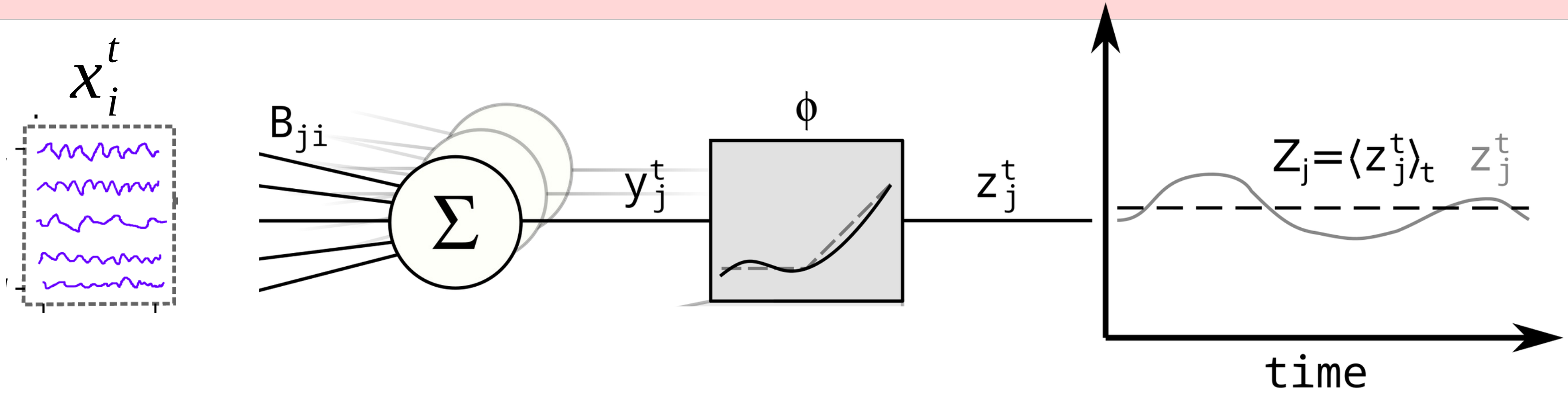
Order-selective perceptron (OSP)



- Decision based on mean **Z**
- Training with target $\bar{\mathbf{Z}}$

Nestler, Helias, Gilson (submitted) *arxiv*

Order-selective perceptron (OSP)



Equations for activation dynamics

$$y^t = \sum_i B_i x_i^t$$

$$z^t = \Phi(y^t) = \alpha_1 y^t + \alpha_2 (y^t - \langle y^t \rangle)^2 + \alpha_3 (y^t - \langle y^t \rangle)^3$$

Nestler, Helias, Gilson (submitted) *arxiv*

Training the nonlinearity

$$z^t = \alpha_1 y^t + \alpha_2 \left(y^t - \langle y^t \rangle \right)^2 + \alpha_3 \left(y^t - \langle y^t \rangle \right)^3$$

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$ after averaging over observation window

cumulants of orders 1, 2 and 3

Nestler, Helias, Gilson (submitted) *arxiv*

Training the nonlinearity

$$z^t = \alpha_1 y^t + \alpha_2 (y^t - \langle y^t \rangle)^2 + \alpha_3 (y^t - \langle y^t \rangle)^3$$

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$ after averaging over observation window

Error $\epsilon = \frac{1}{2} \| Z - \bar{Z} \|^2$ for target $\bar{Z}$ when presenting activity x_i^t

Nestler, Helias, Gilson (submitted) *arxiv*

Training the nonlinearity

$$z^t = \alpha_1 y^t + \alpha_2 \left(y^t - \langle y^t \rangle\right)^2 + \alpha_3 \left(y^t - \langle y^t \rangle\right)^3$$

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$ after averaging over observation window

Error $\epsilon = \frac{1}{2} \| Z - \bar{Z} \|^2$ for target $\bar{Z}$ when presenting activity x_i^t

$$\Delta \alpha_k = - \frac{\partial \epsilon}{\partial Z} \frac{\partial Z}{\partial \alpha_k} = \left(\bar{Z} - Z\right) Y^{'k'}$$ using $$\frac{\partial Z}{\partial \alpha_k} = Y^{'k'}$$

chain rule

Nestler, Helias, Gilson (submitted) *arxiv*

Training the afferent weights using back-propagation

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$

and

$$y^t = \sum_i B_i x_i^t$$

so

$$\frac{\partial Z}{\partial B} = \sum_k \alpha_k \frac{\partial Y^{'k'}}{\partial B}$$

Nestler, Helias, Gilson (submitted) *arxiv*

Training the afferent weights using back-propagation

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$ and $$y^t = \sum_i B_i x_i^t$$

so $$\frac{\partial Z}{\partial B} = \sum_k \alpha_k \frac{\partial Y^{'k'}}{\partial B}$$

we obtain $$\Delta B = -\frac{\partial \epsilon}{\partial Z} \frac{\partial Z}{\partial B} = (\bar{Z} - Z) \sum_k \alpha_k \frac{\partial Y^{'k'}}{\partial B}$$

Nestler, Helias, Gilson (submitted) *arxiv*

Training the afferent weights using back-propagation

$$Z = \alpha_1 Y^{'1'} + \alpha_2 Y^{'2'} + \alpha_3 Y^{'3'}$$ and $$y^t = \sum_i B_i x_i^t$$

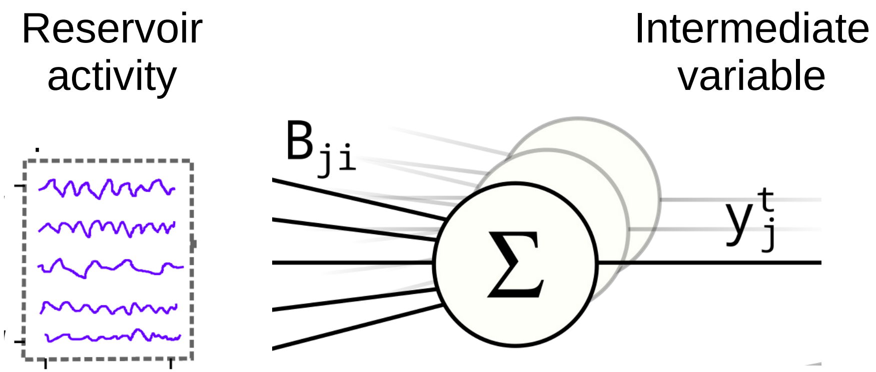
?

so $$\frac{\partial Z}{\partial B} = \sum_k \alpha_k \frac{\partial Y^{'k'}}{\partial B}$$

we obtain $$\Delta B = -\frac{\partial \epsilon}{\partial Z} \frac{\partial Z}{\partial B} = (\bar{Z} - Z) \sum_k \alpha_k \frac{\partial Y^{'k'}}{\partial B}$$

Nestler, Helias, Gilson (submitted) *arxiv*

Derivatives of cumulants with respect to afferent weights



$$x_i^t$$

$$y_j^t = \sum_i B_{ji} x_i^t$$

Linear perceptron

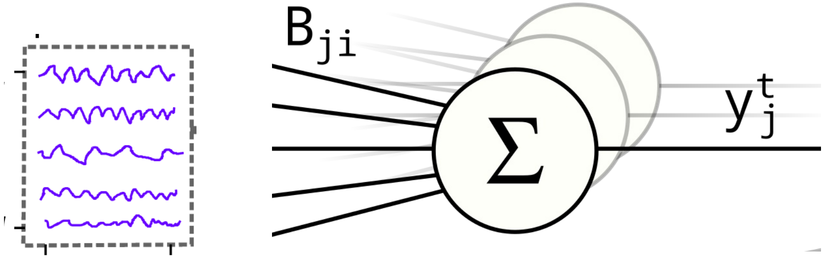
Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Derivatives of cumulants with respect to afferent weights

Reservoir activity

Intermediate variable

Temporal average



$$Y^{'1'} = \langle y^t \rangle = B X^{'1'}$$

$$x_i^t$$

$$y_j^t = \sum_i B_{ji} x_i^t$$

Linear perceptron

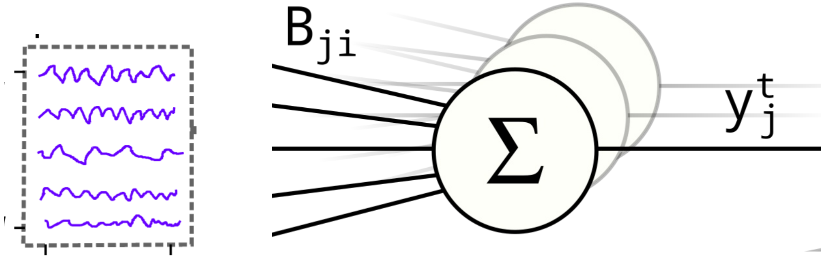
Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Derivatives of cumulants with respect to afferent weights

Reservoir activity

Intermediate variable

Temporal average



$$Y^{'1'} = \langle y^t \rangle = B\, X^{'1'}$$

$$x_i^t$$

$$y_j^t = \sum_i B_{ji}\, x_i^t$$

Linear perceptron

$$\boxed{\frac{\partial Y^{'1'}}{\partial B} = X^{'1'}}$$

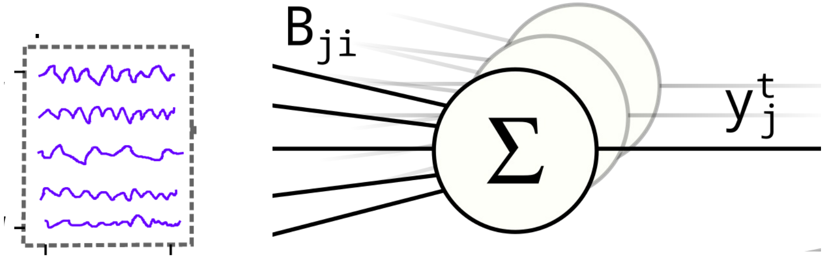
Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Derivatives of cumulants with respect to afferent weights

Reservoir activity

Intermediate variable

Temporal covariance



$$Y^{'2'} = \langle y^t \, y^t \rangle = B \, X^{'2'} \, B^T$$

after centering

$$x_i^t$$

$$y_j^t = \sum_i B_{ji} \, x_i^t$$

Linear perceptron

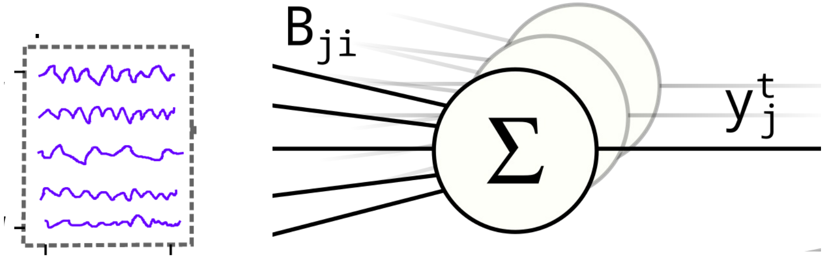
Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Derivatives of cumulants with respect to afferent weights

Reservoir activity

Intermediate variable

Temporal covariance



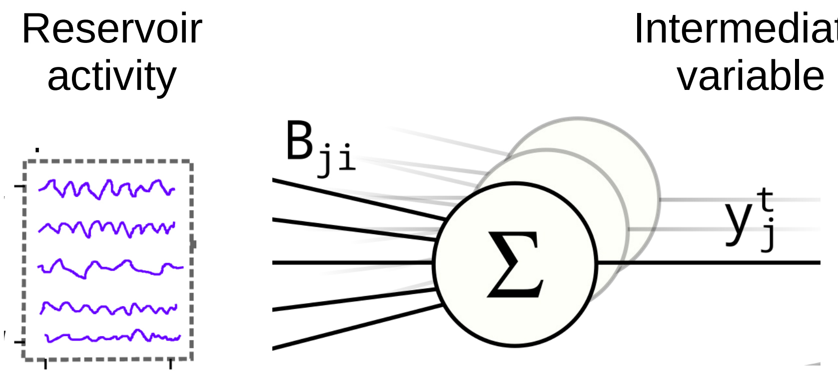
$$Y^{'2'} = \langle y^t \, y^t \rangle = B \, X^{'2'} \, B^T$$

$$\frac{\partial Y^{'2'}}{\partial B} = B \, X^{'2'} + X^{'2'} \, B^T$$

$$x_i^t$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Derivatives of cumulants with respect to afferent weights



x_i^t

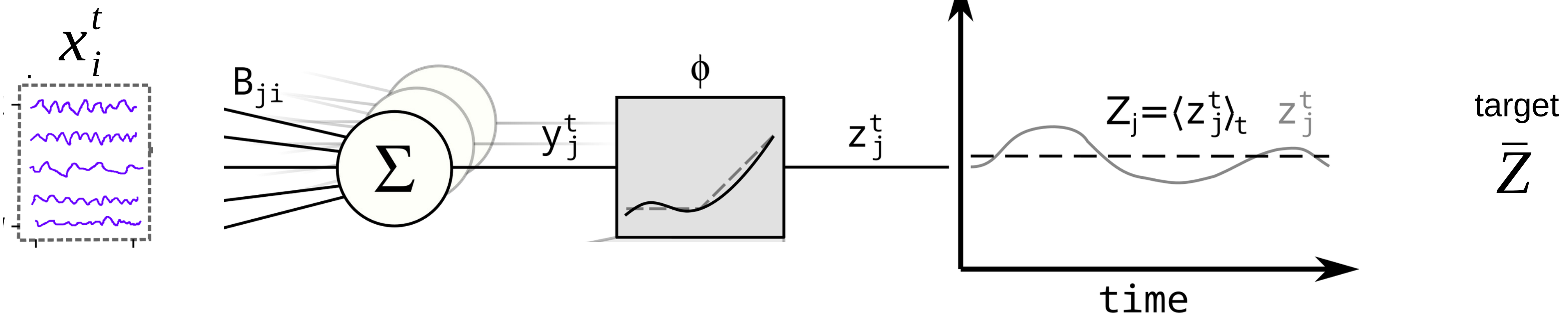
Third-order covariance

$$Y^{'3'} = \langle y^t y^t y^t \rangle$$
$$= B \odot^1 B \odot^2 B \odot^3 X^{'3'}$$

$$\frac{\partial Y^{'3'}}{\partial B} = B \odot^1 B \odot^2 X^{'3'} + \dots$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Putting everything together:
Automated selection of relevant input cumulant for classification



$$\Delta\alpha_k = (\bar{Z} - Z)\, Y^{'k'}$$

$$\Delta B = (\bar{Z} - Z) \sum_k \alpha_k [B, \ldots, B] \odot X^{'k'}$$

depends on k

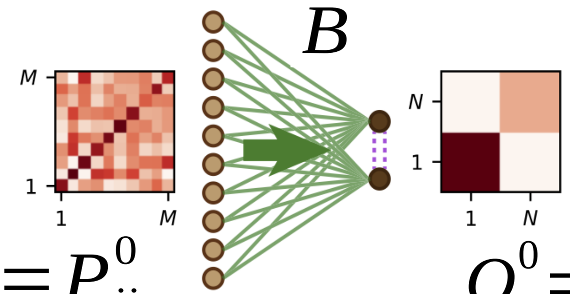
Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Weight update in linear network to tune output covariance

Covariance mapping

Target

Error (matrix norm)



$$\langle x_i^t x_j^t \rangle = P_{ij}^0$$

after centering

$$Q^0 = B P^0 B^T$$

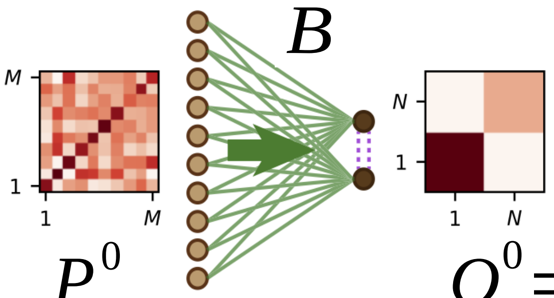
Multivariate output here!!!

Weight update

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Weight update in linear network to tune output covariance

Covariance mapping



P^0

Target

Error (matrix norm)

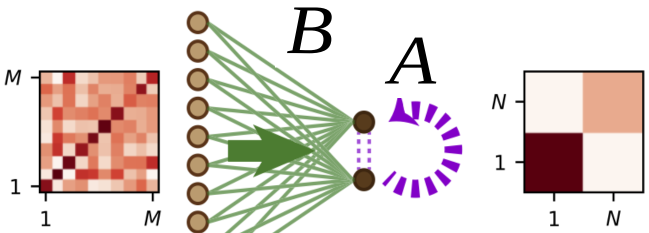
$$Q^0 = B\,P^0\,B^T \Leftrightarrow \bar{Q}^0 \qquad \epsilon = ||\bar{Q}^0 - Q^0||^2$$

Weight update $$\Delta B = -\frac{\partial \epsilon}{\partial Q^0} \frac{\partial Q^0}{\partial B}$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Naturally extends to recurrent networks

Covariance mapping



$$\langle x_i^t x_j^t \rangle = P_{ij}^0 \qquad Q^0 = \ldots$$

$$\langle x_i^t x_j^{t-1} \rangle = P_{ij}^1 \qquad Q^1 = \ldots$$

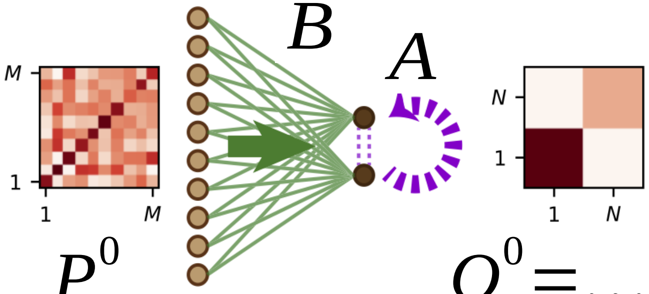
$$y_j^t = \sum_i A_{ji} y_i^{t-1} + \sum_i B_{ji} x_i^t$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Naturally extends to recurrent networks

Covariance mapping

Targets



P^0 $\quad Q^0 = \ldots \quad \bar{Q}^0$

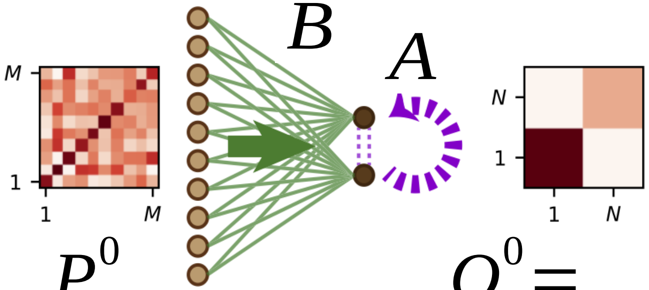
P^1 $\quad Q^1 = \ldots \quad \bar{Q}^1$

$$y_j^t = \sum_i A_{ji} y_i^{t-1} + \sum_i B_{ji} x_i^t$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Naturally extends to recurrent networks

Covariance mapping



P^0 $\quad Q^0 = \dots$

P^1 $\quad Q^1 = \dots$

Targets

$\bar{Q}^0$

$\bar{Q}^1$

Error (matrix norm)

$$\epsilon = ||\bar{Q}^0 - Q^0||^2 + ||\bar{Q}^1 - Q^1||^2$$

Weight updates

$$\Delta B = -\frac{\partial \epsilon}{\partial Q^0}\frac{\partial Q^0}{\partial B} - \frac{\partial \epsilon}{\partial Q^1}\frac{\partial Q^1}{\partial B}$$

$$\Delta A = -\frac{\partial \epsilon}{\partial Q^0}\frac{\partial Q^0}{\partial A} - \frac{\partial \epsilon}{\partial Q^1}\frac{\partial Q^1}{\partial A}$$

Gilson*, Dahmen*, ..., Helias *PLoS Comput Biol* 2020

Summary on theory

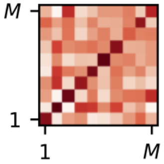
- Training nonlinearity and afferent input weights to tune output mean
- Training afferent and recurrent weights in linear network to tune output covariances (also extends to higher orders)

- Now: tune output covariances when including nonlinearity in network

Capacity of covariance perceptron

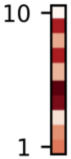
- How many binary patterns can be categorized into 2 groups?
- Feedforward linear perceptron

covariances



P^0

means



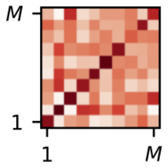
X

Dahmen, Gilson, Helias *J Phys A* 2020

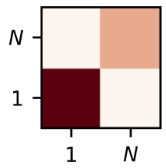
Capacity of covariance perceptron

- How many binary patterns can be categorized into 2 groups?
- Feedforward linear perceptron

covariances

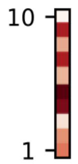


$$P^0$$



$$Q^0 = B\,P^0\,B^T$$

means



$$X$$

$$Y = B\,X$$

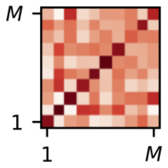
Dahmen, Gilson, Helias *J Phys A* 2020

Capacity of covariance perceptron

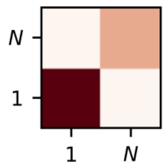
- How many binary patterns can be categorized into 2 groups?
- Feedforward linear perceptron

Replica method

covariances

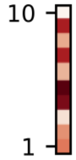


P^0



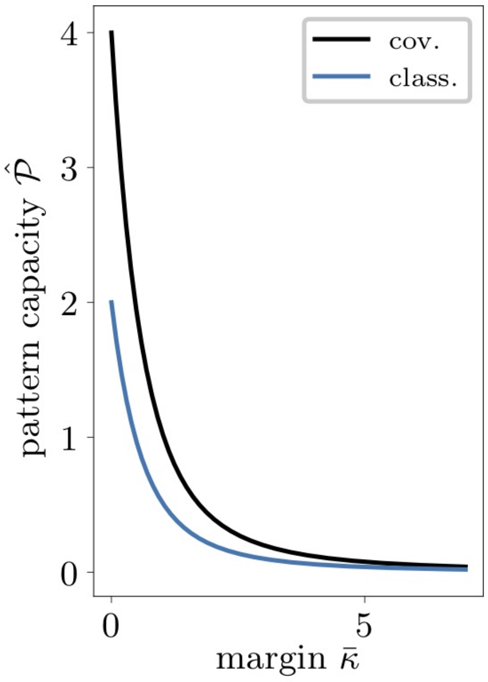
$Q^0 = B\,P^0\,B^T$

means



X

$Y = B\,X$



Dahmen, Gilson, Helias *J Phys A* 2020