Multi-way Feature Selection for ECoG-based Hand Movement Prediction

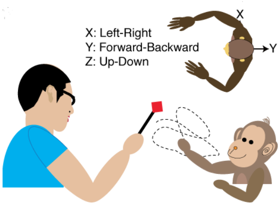
Anastasia Motrenko, Vadim Strijov

Moscow Institute of Physics and Technology

May 25, 2017

Neurotycho data, foodtracking task

A monkey is tracking food rewards with the hand contralateral to the implant side. The experimenter demonstrated foods at random locations at a distance of 20 cm for the monkey at random time intervals 3-4 times per minute, and the monkey grasped the foods



- Subdural (32 electrodes): 2 monkeys, 3 and 5 records, taken within 7 months.
- Each record measures about 1000 seconds with ECoG and motion data (wrists, elbows and shouders) sampled at 1KHz and 120Hz, respectively.

Problem statement: movement prediction

Inputs: multivariate time series $\mathbf{s}(t) \in \mathbb{R}^{N_{\text{ch}}}$ — voltage measurements for each channel $1, \ldots, N_{\text{ch}}$.
Targets: multivariate time series $\mathbf{y}(t) \in \mathbb{R}^3$ with 3D limb coordinates.

The goal us to reconstruct $\mathbf{y}(t)$ from $\mathbf{s}(t), \ldots \mathbf{s}(t - \Delta t)$.

The time series are converted to the data sample $(\underline{\mathbf{D}}, \mathbf{Y})$:

$$\underline{\mathbf{D}} \in \mathbb{R}^{T \times F \times N_{\text{ch}} \times M}, \ D_{(m,:,:,:)} = \underline{\mathbf{X}}_m, \quad \mathbf{Y} = [\mathbf{y}_1^{\mathsf{T}}, \ldots, \mathbf{y}_M^{\mathsf{T}}]^{\mathsf{T}},$$

such that $\mathbf{y}_m = \mathbf{y}(t_m)$ and $\underline{\mathbf{X}}_m \in \mathbb{R}^{T \times F \times N_{\text{ch}}}$ is a three-way matrix, which stores time-frequency features extracted from the time series $[s_n(t_m - \Delta t), \ldots, s_n(t_m))]$ along all channels n, $n = 1, \ldots, N_{\text{ch}}$.

The reconstructed trajectory $\hat{\mathbf{Y}}$ approximates the real $\mathbf{Y}$ as a linear combination of features:

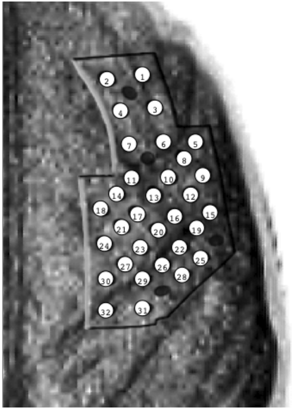
$$\hat{\mathbf{y}}_m = \text{vec}\big(\underline{\mathbf{X}}_m\big)^{\mathsf{T}} \hat{\mathbf{w}},$$

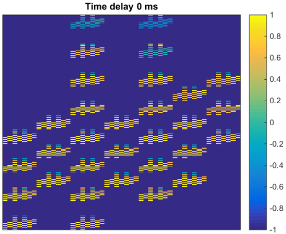
where the weight vector $\hat{\mathbf{w}} \in \mathbb{R}^{T \cdot F \cdot N_{\text{ch}} \times 3}$ minimize the squared sum of residues:

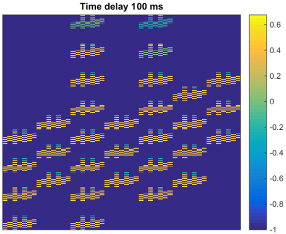
$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} ||\hat{\mathbf{Y}} - \mathbf{Y}||_2^2.$$

Feature extraction: time-domain features

Correlations between channels in time domain:

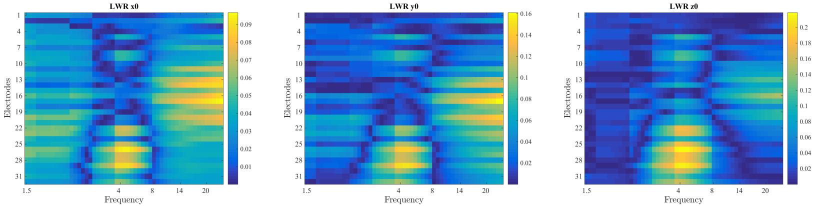






Problem statement: feature selection

Absolute values of cross-correlation between ECoG and target time series in (left wrist) frequency domain. No time delay $\tau = 0$.



Problem statement: feature selection

Let $\mathbf{X} \in \mathbb{R}^{M \times T \cdot F \cdot N_{\text{ch}}}$ denote vectorized feature matrix $\underline{\mathbf{D}} \in \mathbb{R}^{T \times F \times N_{\text{ch}} \times M}$:

$$\mathbf{X} = [\text{vec}(\underline{\mathbf{X}}_1), \dots, \text{vec}(\underline{\mathbf{X}}_M)]^{\mathsf{T}} =$$

$$[\dots, \chi_{(i,j,n)}, \dots], \; (i, j, n) \in \{1, \dots, T\} \times \{1, \dots, F\} \times \{1, \dots, N_{\text{ch}}\}.$$

Indicator variable $\underline{\mathbf{A}} \in \mathbb{R}^{T \times F \times N_{\text{ch}}}$ encodes inclusions of features $\chi_{(i,j,n)}$ into the dataset and the corresponding two-way feature matrix $\mathbf{X}_{\underline{\mathbf{A}}}$:

$$\mathbf{X}_{\underline{\mathbf{A}}} = [\dots, \chi_{(i,j,n)}, \dots], \text{ such that } \underline{\mathbf{A}}_{ijn} = 1.$$

Feature selection problem is formulated the following way:

$$\underline{\mathbf{A}} = \underset{\underline{\mathbf{A}} \in \mathbb{R}^{T \times F \times N_{\text{ch}}}}{\arg\min} \; \mathcal{L}\left(\mathbf{X}_{\underline{\mathbf{A}}} \mathbf{w}_{\underline{\mathbf{A}}}, \mathbf{Y}\right),$$

where $\mathcal{L}(\hat{Y}, \mathbf{Y})$ is some loss function and $\mathbf{w}_{\underline{\mathbf{A}}}$ minimizes quadratic loss for $\mathbf{X}_{\underline{\mathbf{A}}}$.

Quadratic Programming Feature Selection (QPFS)

The feature selection problem is formulated as quadratic programming problem

$$\mathbf{a} = \mathop{\arg\min}_{\mathbf{a}\in\{0,1\}^N} \left(\mathbf{a}^\mathsf{T}\mathbf{Q}\mathbf{a} - \mathbf{b}^\mathsf{T}\mathbf{a}\right),$$

where q_{ij} entry of matrix $\mathbf{Q} \in \mathbb{R}^{N\times N}$ quantifies *similarity* between i-th and j-th features, say

$$q_{ij} = |\text{corr}(\boldsymbol{\chi}_i, \boldsymbol{\chi}_j)|.$$

Here $\boldsymbol{\chi}_i$, $\boldsymbol{\chi}_j$ denote columns of the design matrix $\mathbf{X}$. Similarly, element b_i, which is referred to as *relevance* of the $i-$th feature, quantifies similarity between $\boldsymbol{\chi}_i$ and the target $\mathbf{Y}$:

$$b_i = \frac{1}{3}\sum_{n=1}^{3} |\text{corr}(\boldsymbol{\chi}_i, \mathbf{y}_n)|.$$

Other options:

- mutual information $\text{MI}(\boldsymbol{\chi}_i, \boldsymbol{\chi}_j)$ for similarity q_{ij} and
- normalized feature significance for relevance b_i.

Multi-way QPFS

Design matrix $\mathbf{X} \in \mathbb{R}^{M \times n}$	$\to$	$\underline{\mathbf{D}} \in \mathbb{R}^{M \times n_1 \times n_2 (\times n_3)}$
Feature vector $\chi_i \in \mathbb{R}^n$	$\to$	$\underline{\mathbf{X}}_{(}\ :, i_1, i_2, i_3) \in \mathbb{R}^{n_1 n_2 n_3}$
Similarity matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$	$\to$	similarity matrices $\mathbf{Q}_1 \in \mathbb{R}^{n_1 \times n_1}$, $\mathbf{Q}_2 \in \mathbb{R}^{n_2 \times n_2}$, $\mathbf{Q}_3 \in \mathbb{R}^{n_3 \times n_3}$ for each mode
Relevance vector $\mathbf{b} \in \mathbb{R}^n$	$\to$	multi-way matrix $\underline{\mathbf{B}} \in \mathbb{R}^{n_1 \times n_2 (\times n_3)}$
Structure variable $\mathbf{a} \in \mathbb{R}^n$	$\to$	$\underline{\mathbf{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$

Feature selection problem reformulates as follows:

$$\underline{\mathbf{A}} = \mathop{\arg\min}_{\underline{\mathbf{A}} \in \{0,1\}^{n_1 \times n_2 \times n_3}} \sum_{d=1}^{3} \mathcal{Q}(\underline{\mathbf{A}}; \mathbf{Q}_d) - \mathcal{Q}(\underline{\mathbf{A}}; \underline{\mathbf{B}}),$$

where

$\mathbf{a}^{\mathsf{T}}\mathbf{Q}\mathbf{a}$	$\to$	$\mathcal{Q}(\underline{\mathbf{A}}; \mathbf{Q}_d) = (\underline{\mathbf{A}} \times_1 \mathbf{Q}_d) * \underline{\mathbf{A}} \times_1 \mathbf{1}_{n_1} \times_2 \mathbf{1}_{n_2} \times_3 \mathbf{1}_{n_3}$,
$\mathbf{b}^{\mathsf{T}}\mathbf{a}$	$\to$	$\mathcal{Q}(\underline{\mathbf{A}}; \underline{\mathbf{B}}) = \underline{\mathbf{B}} * \underline{\mathbf{A}} \times_1 \mathbf{1}_{n_1} \times_2 \mathbf{1}_{n_2} \times_3 \mathbf{1}_{n_3}$.

- $*$ stands for element-wise product: $[\underline{\mathbf{A}} * \underline{\mathbf{B}}]_{ijk} = a_{ijk} b_{ikj}$.
- $\underline{\mathbf{A}} \times_1 \mathbf{B} \in \mathbb{R}^{m \times n_2 \times n_3}$ denotes inner product of multi-way matrix $\underline{\mathbf{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$ to matrix $\mathbf{B} \in \mathbb{R}^{m \times n_1}$: $[\underline{\mathbf{A}} \times_1 \mathbf{B}]_{ijk} = \sum_{i'} a_{i'jk} b_{ii'}$
- operation $\underline{\mathbf{A}} \times_1 \mathbf{1}_{n_1} \times_2 \mathbf{1}_{n_2} \times_3 \mathbf{1}_{n_3}$ is equivalent to summation over all entries of $\underline{\mathbf{A}}$:

$$\underline{\mathbf{A}} \times_1 \mathbf{1}_{n_1} \times_2 \mathbf{1}_{n_2} \times_3 \mathbf{1}_{n_3} = \sum_{i=1}^{n_1} \sum_{j=1}^{n_3} \sum_{k=1}^{n_3} a_{ijk}.$$

Multi-way QPFS

Since $\underline{\mathbf{A}}$ is binary, an exact low-rank decomposition is possible:

$$\underline{\mathbf{A}} = \sum_{r=1}^{R} \mathbf{a}_1^{(r)} \circ \mathbf{a}_2^{(r)} \circ \mathbf{a}_3^{(r)}, \quad \mathbf{a}_1^{(r)} \in \mathbb{R}^{n_1},\ \mathbf{a}_2^{(r)} \in \mathbb{R}^{n_2},\ \mathbf{a}_3^{(r)} \in \mathbb{R}^{n_3}.$$

This allows to rewrite the objective $\mathcal{Q}(\underline{\mathbf{A}})$ as

$$\begin{aligned}\mathcal{Q}(\underline{\mathbf{A}}) = \sum_{r=1}^{R} \quad & ||\mathbf{a}_2^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_1^{(r)\mathsf{T}} \mathbf{Q}_1 \mathbf{a}_1^{(r)} + \\ + \quad & ||\mathbf{a}_1^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_2^{(r)\mathsf{T}} \mathbf{Q}_2 \mathbf{a}_2^{(r)} + \\ + \quad & ||\mathbf{a}_1^{(r)}||_2^2 \cdot ||\mathbf{a}_2^{(r)}||_2^2 \cdot \mathbf{a}_3^{(r)\mathsf{T}} \mathbf{Q}_3 \mathbf{a}_3^{(r)} + \\ + \quad & \underline{\mathbf{B}} \times_1 \mathbf{a}_1^{(r)} \times_2 \mathbf{a}_2^{(r)} \times_3 \mathbf{a}_3^{(r)}.\end{aligned}$$

and solve M-QPFS problem via alternate approach so that at each step a quadratic programming with problem is solved.

Multi-way QPFS

$$\underline{\mathbf{A}} = \arg\min_{\mathbf{a}_d^{(r)}} \sum_{r=1}^{R} \quad ||\mathbf{a}_2^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_1^{(r)\mathsf{T}} \mathbf{Q}_1 \mathbf{a}_1^{(r)} +$$
$$+ \quad ||\mathbf{a}_1^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_2^{(r)\mathsf{T}} \mathbf{Q}_2 \mathbf{a}_2^{(r)} +$$
$$+ \quad ||\mathbf{a}_1^{(r)}||_2^2 \cdot ||\mathbf{a}_2^{(r)}||_2^2 \cdot \mathbf{a}_3^{(r)\mathsf{T}} \mathbf{Q}_3 \mathbf{a}_3^{(r)} +$$
$$+ \quad \underline{\mathbf{B}} \times_1 \mathbf{a}_1^{(r)} \times_2 \mathbf{a}_2^{(r)} \times_3 \mathbf{a}_3^{(r)}.$$

Let $\boldsymbol{\alpha}_d = [\mathbf{a}_d^{(1)\mathsf{T}}, \dots, \mathbf{a}_d^{(R)\mathsf{T}}]^\mathsf{T} \in \mathbb{R}^{nR}$ for $d = 1, 2, 3$ and $\boldsymbol{\alpha}^{(0)} = \mathbf{1}_{n_d R}$ be the initial approximation of $\boldsymbol{\alpha}_d$.
Fix $\boldsymbol{\alpha}_2^{(k-1)}$, $\boldsymbol{\alpha}_3^{(k-1)}$ and solve the following problem with respect to $\boldsymbol{\alpha}_1$:

$$\boldsymbol{\alpha}_1^{(k)} = \arg\min_{\boldsymbol{\alpha} \in \{0,1\}^{nR}} \boldsymbol{\alpha}_1^\mathsf{T} \left(\tilde{\mathbf{Q}}_1^{(k-1)} + \tilde{\mathbf{I}}_1^{(k-1)} \right) \boldsymbol{\alpha}_1 + \tilde{\mathbf{B}}_1^{(k-1)} \boldsymbol{\alpha}_1,$$

where $\tilde{\mathbf{Q}}_1^{(k)}$ and $\tilde{\mathbf{I}}_1^{(k-1)}$ are block-diagonal with $r-$th blocks $\tilde{\mathbf{Q}}_1^{(k,r)}$ and $\tilde{\mathbf{I}}_1^{(k-1)}$:

- $\tilde{\mathbf{Q}}_1^{(k,r)} = ||\mathbf{a}_2^{(k,r)}||_2^2 \cdot ||\mathbf{a}_3^{(k,r)}||_2^2 \mathbf{Q}_1,$
- $\tilde{\mathbf{I}}_1^{(k-1)} = (||\mathbf{a}_3^{(k,r)}||_2^2 \cdot \mathbf{a}_2^{(k,r)\mathsf{T}} \mathbf{Q}_2 \mathbf{a}_2^{(k,r)} + ||\mathbf{a}_2^{(k,r)}||_2^2 \cdot \mathbf{a}_3^{(k,r)\mathsf{T}} \mathbf{Q}_3 \mathbf{a}_3^{(r)}) \mathbf{I}_{n_1},$
- $\tilde{\mathbf{B}}_1^{(k)} = [\tilde{\mathbf{B}}_1^{(k,1)\mathsf{T}}, \dots, \tilde{\mathbf{B}}_1^{(k,R)\mathsf{T}}]^\mathsf{T}, \quad \tilde{\mathbf{B}}_1^{(k,r)} = \underline{\mathbf{B}} \times_2 \mathbf{a}_2^{(k,r)} \times_3 \mathbf{a}_3^{(k,r)}.$

Multi-way QPFS

$$\underline{\mathbf{A}} = \arg\min_{\mathbf{a}_d^{(r)}} \sum_{r=1}^{R} \quad ||\mathbf{a}_2^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_1^{(r)\mathsf{T}} \mathbf{Q}_1 \mathbf{a}_1^{(r)} +$$
$$+ \quad ||\mathbf{a}_1^{(r)}||_2^2 \cdot ||\mathbf{a}_3^{(r)}||_2^2 \cdot \mathbf{a}_2^{(r)\mathsf{T}} \mathbf{Q}_2 \mathbf{a}_2^{(r)} +$$
$$+ \quad ||\mathbf{a}_2^{(r)}||_2^2 \cdot ||\mathbf{a}_1^{(r)}||_2^2 \cdot \mathbf{a}_3^{(r)\mathsf{T}} \mathbf{Q}_3 \mathbf{a}_3^{(r)} +$$
$$+ \quad \underline{\mathbf{B}} \times_1 \mathbf{a}_1^{(r)} \times_2 \mathbf{a}_2^{(r)} \times_3 \mathbf{a}_3^{(r)}.$$

Now $[\mathbf{a}_1^{(1)\mathsf{T}}, \dots, \mathbf{a}_1^{(R)\mathsf{T}}]^\mathsf{T} = \boldsymbol{\alpha}_1^{(k)}$.

Fix $\boldsymbol{\alpha}_1^{(k)}$, $\boldsymbol{\alpha}_3^{(k-1)}$ and solve the following problem with respect to $\boldsymbol{\alpha}_2$:

$$\boldsymbol{\alpha}_2^{(k)} = \arg\min_{\boldsymbol{\alpha} \in \{0,1\}^{nR}} \boldsymbol{\alpha}_2^\mathsf{T} \left(\tilde{\mathbf{Q}}_2^{(k-1)} + \tilde{\mathbf{I}}_2^{(k-1)} \right) \boldsymbol{\alpha}_2 + \tilde{\mathbf{B}}_2^{(k-1)} \boldsymbol{\alpha}_2,$$

where $\tilde{\mathbf{Q}}_2^{(k)}$ and $\tilde{\mathbf{I}}_2^{(k-1)}$ are block-diagonal with $r-$th blocks $\tilde{\mathbf{Q}}_2^{(k,r)}$ and $\tilde{\mathbf{I}}_2^{(k-1)}$:

- $\tilde{\mathbf{Q}}_2^{(k,r)} = ||\mathbf{a}_1^{(k,r)}||_2^2 \cdot ||\mathbf{a}_3^{(k,r)}||_2^2 \mathbf{Q}_2,$
- $\tilde{\mathbf{I}}_2^{(k-1)} = (||\mathbf{a}_3^{(k,r)}||_2^2 \cdot \mathbf{a}_1^{(k,r)\mathsf{T}} \mathbf{Q}_1 \mathbf{a}_1^{(k,r)} + ||\mathbf{a}_1^{(k,r)}||_2^2 \cdot \mathbf{a}_3^{(k,r)\mathsf{T}} \mathbf{Q}_3 \mathbf{a}_3^{(r)}) \mathbf{I}_{n_1},$
- $\tilde{\mathbf{B}}_2^{(k)} = [\tilde{\mathbf{B}}_2^{(k,1)\mathsf{T}}, \dots, \tilde{\mathbf{B}}_2^{(k,R)\mathsf{T}}]^\mathsf{T}, \quad \tilde{\mathbf{B}}_2^{(k,r)} = \underline{\mathbf{B}} \times_1 \mathbf{a}_1^{(k,r)} \times_3 \mathbf{a}_3^{(k,r)}.$

Repeat for $\boldsymbol{\alpha}_3$.

Linear relaxation

QPFS is the integer optimization problem, not convex.

To allow for efficient solution, we have to relax the constraints

$$\underline{\mathbf{A}} \in \{0, 1\}^{n_1 \times n_2 \times n_3} \quad \to \quad \underline{\mathbf{A}} \in [0, 1]^{n_1 \times n_2 \times n_3}.$$

After the solution $\hat{\underline{\mathbf{A}}}$ of the relaxed problem is found, $\hat{\underline{\mathbf{A}}}$ is thresholded

$$\underline{\mathbf{A}}(\epsilon) = [a_{ijk}], \quad a_{ijk} = \begin{cases} 1 \text{ if } \hat{a}_{ijk} \geq \epsilon, \\ 0 \text{ otherwise.} \end{cases}$$

to select a number of features $\mathbf{X}_{\underline{\mathbf{A}}}$.

Setting various threshold values ϵ, one obtains various active sets of features $\mathbf{X}_{\underline{\mathbf{A}}(\epsilon)}$.

Feature extraction: frequency-domain features

To obtain $T \times F$ features in time-frequency domain for each of N_{ch}

1. select M time points $t_1, \dots, t_M$ with time step δt;
2. select F basic frequencies (scales) f_j, $j = 1, \dots, F$;
3. apply Morlet wavelet transform to all $s_n(t)$, $n = 1, \dots, N_{\text{ch}}$ at each center $t_1 \leq t_i \leq t_M$ and scale f_j, $j = 1, \dots, F$:

$$W_{ijn} = \frac{1}{\sqrt{|f_j|}} \sum_{t \leq t_M} \psi\left(\frac{t - t_i}{f_j}\right) s_n(t).$$

The feature matrix $\underline{\mathbf{X}}_m$ comprises information about the time series $\mathbf{s}(t)$ across the time period $t_m - \Delta < t \leq t_m$.

Feature extraction

2D dataset:

- the time-delayed ($\tau = 0.65s$) ECoG time series
- wavelet coefficients:

$$\underline{\mathbf{X}}_m \in \mathbb{R}^{F \times N_{\text{ch}}}, \quad \underline{\mathbf{X}}_{mjn} = \begin{cases} s_n(t_m + \tau), j = 1, \\ W_{mjn} \text{ for } j = 2, \dots, F+1, \end{cases} \quad n = 1, \dots, N_{\text{ch}}.$$

- The time series were downsampled the data by the factor of 10.
- Frequency bands: 0.5–8Hz with 0.5Hz step, 9–18Hz with 3Hz and 20-45 with 5Hz step.

To create the data set we used the data from 5 to 950 seconds with time step $\delta t = 0.05s$.
Resulting design matrix: $\underline{\mathbf{D}} \in \mathbf{18901} \times \mathbf{32} \times \mathbf{27}$.

Feature extraction

3D dataset contains 3-way features with no time delay.

3D features explicitly include local history $\Delta_m = [t_m - \Delta t, t_m]$ of wavelet coefficients.

To construct 3D dataset for $t_1, \ldots, t_M$ split the time range Δ_m into T consecutive intervals δt_i, $i = 1, \ldots, T$.

For n-th electrode in $1, \ldots, N$ the (i, j, n)-th element of 3-way matrix $\underline{\mathbf{X}}_m \in \mathbb{R}^{T \times F \times N_{\text{ch}}}$ is given by averaging $W_{i'jn}$ over δ_i:

$$X_{mijn} = \frac{1}{|\delta t_i|} \sum_{i' \colon t_{i'} \in \delta t_i} W_{i'jn}.$$

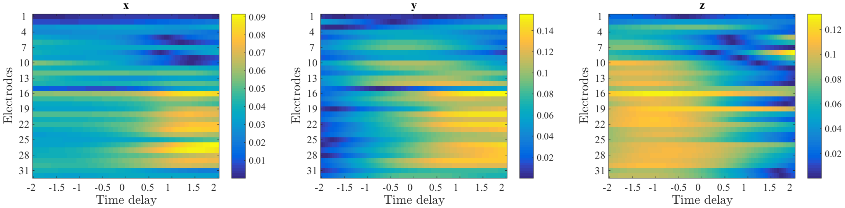
Scalogram features were computed without downsampling with the following parameters: duration of local history time segment $\Delta t = 1s$ with step $\delta t = 0.05s$, $T = 10$, $F = 20$. The frequencies were chosen logarithmically spaced in the range 10 – 500 Hz.
Resulting design matrix: $\underline{\mathbf{D}} \in \mathbf{18901} \times \mathbf{10} \times \mathbf{15} \times \mathbf{32}$.

The optimum latency value is chosen to maximize absolute linear cross-correlation between ECoG $\mathbf{s}(t)$ and target $\mathbf{y}(t)$ time series:

$$\tau_n^* = \underset{\tau \in [\tau_{\min}, \tau_{\max}]}{\arg\max} \frac{\left|\sum_{i=1}^{m} s_n(t_i + \tau) y(t_i)\right|}{\sqrt{\sum_{i=1}^{m} s_n(t_i + \tau) s_n(t_i + \tau)} \sqrt{\sum_{i=1}^{m} y(t_i) y(t_i)}},$$

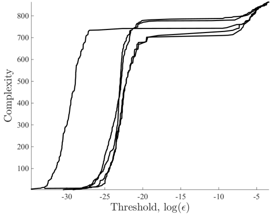
where $y(t)$ is the target time series for a given marker and dimension, and $s_n(t)$ is the ECoG time series for a given electrode.

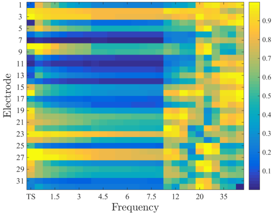
Absolute values of cross-correlation between ECoG and target time series (left wrist) in time domain.



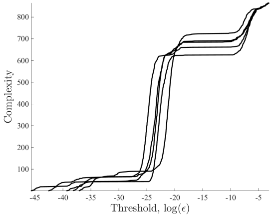
QPFS vs M-QPFS (2D)

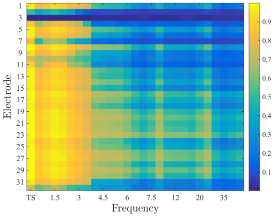
QPFS:





Multi-way QPFS:



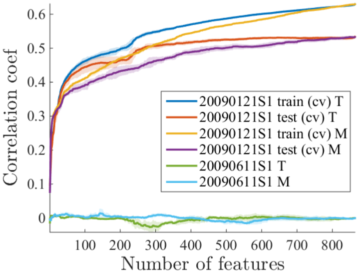


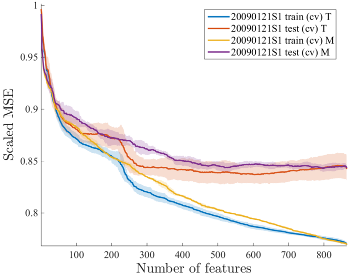
Left: Complexity by the threshold value ϵ.

Right: Ratio of times (i, j)–th feature was selected into active feature set (averaged by threshold values and 5 cross-validation splits).

QPFS vs M-QPFS (2D)

Forecasting quality for the feature sets, defined by (M-)QPFS, by complexity. Time delay $\tau = 0.65s$





Left: correlation coefficient between forecasted hand trajectory $\hat{\mathbf{Y}}$ and the real trajectory $\mathbf{Y}$ (left wrist, monkey A)

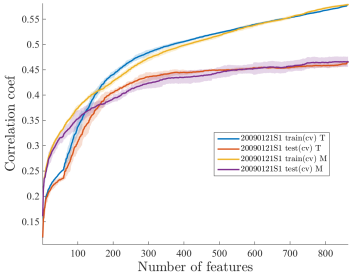
$$\mathrm{corr}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{\mathrm{cov}(\hat{\mathbf{y}}, \mathbf{y})}{\sqrt{\mathrm{cov}(\hat{\mathbf{y}}, \hat{\mathbf{y}})\mathrm{cov}(\mathbf{y}, \mathbf{y})}}$$

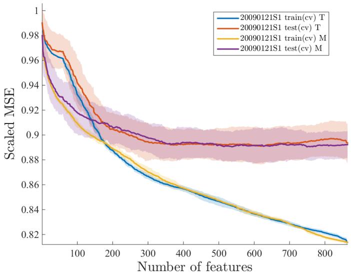
Right: scaled MSE

$$\mathrm{sMSE}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{\sum_{m=1}^{M} ||\hat{\mathbf{y}}_m - \mathbf{y}_m||_2}{\sum_{m=1}^{M} ||\bar{\mathbf{y}} - \mathbf{y}_m||_2}.$$

QPFS vs M-QPFS (2D)

Forecasting quality for the feature sets, defined by (M-)QPFS, by complexity. No time delay, $\tau = 0s$

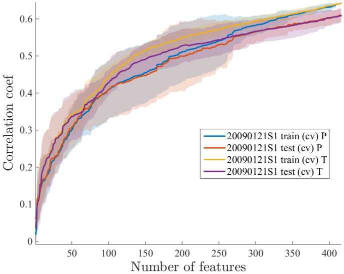


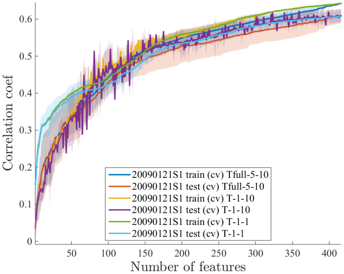


The quality is measured as the correlation coefficient between forecasted hand trajectory and real trajectory (left wrist, monkey A).

Tensor decompositions in M-QPFS (2D)

Left: Comparison of QPFS performance with similarity computed with PARAFAC or Tucker decomposition.
Right: Results with Tucker decomposition, various parameters.

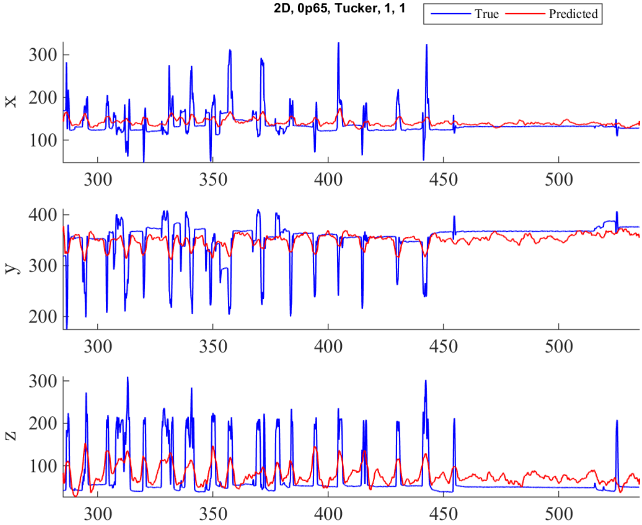


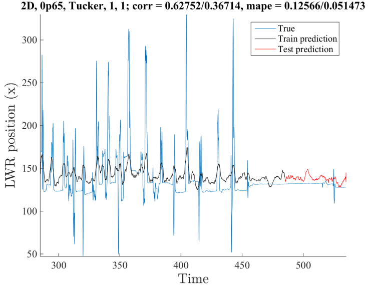


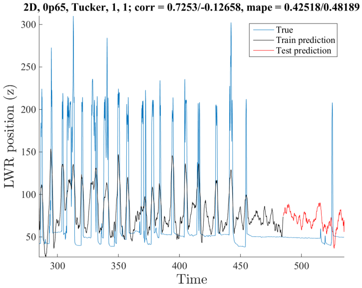
Forecasting quality is measured as the correlation coefficient between the forecasted trajectory and the real trajectory (left wrist, monkey A)

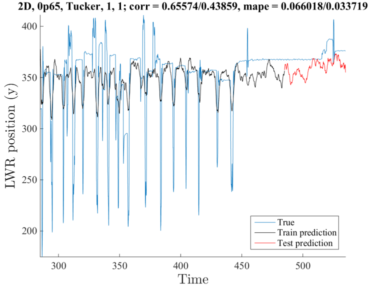
Example of forecast (2D)

50 best (according to M-QPFS) features. Predicted trajectories are smoothed by 2.5s window.



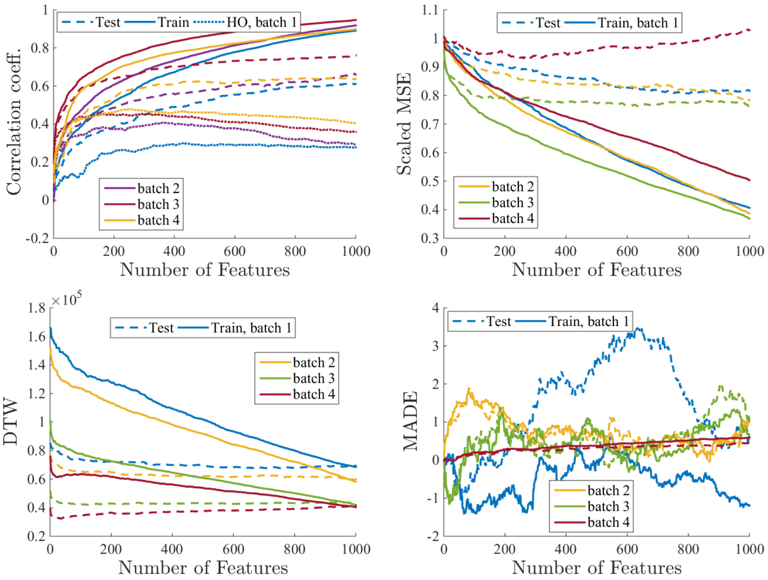




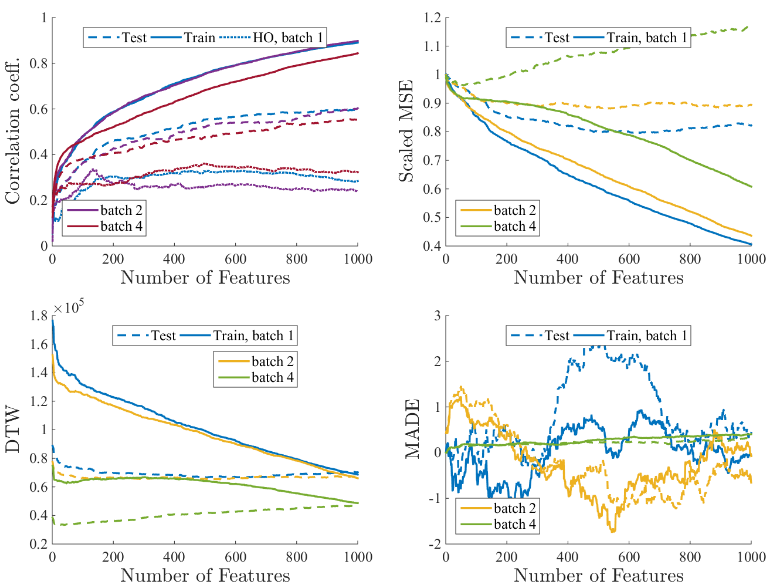


Forecasting quality decreases on intervals of “stillness”.
Multi-modelling?

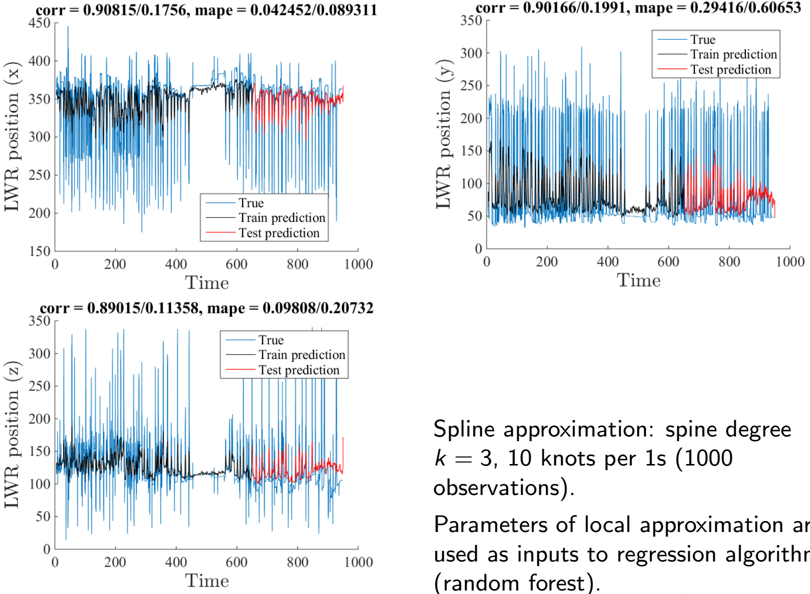
Batch M-QPFS for 3D dataset



Batch QPFS for 3D dataset



Beyond scalograms: spline approximation + random forests



Spline approximation: spine degree $k = 3$, 10 knots per 1s (1000 observations).

Parameters of local approximation are used as inputs to regression algorithm (random forest).