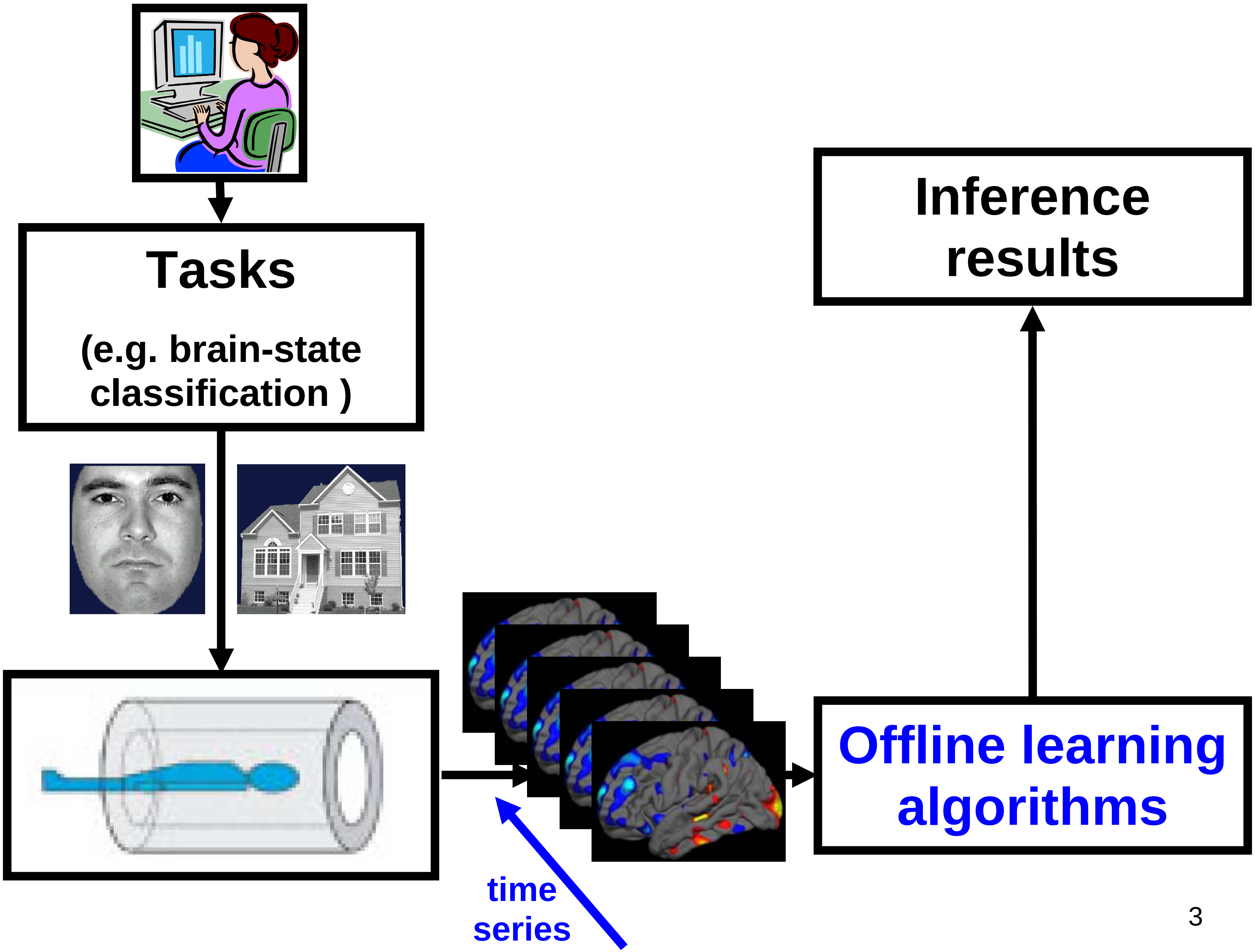
Real-time Conjugate Gradients for Online fMRI Classification

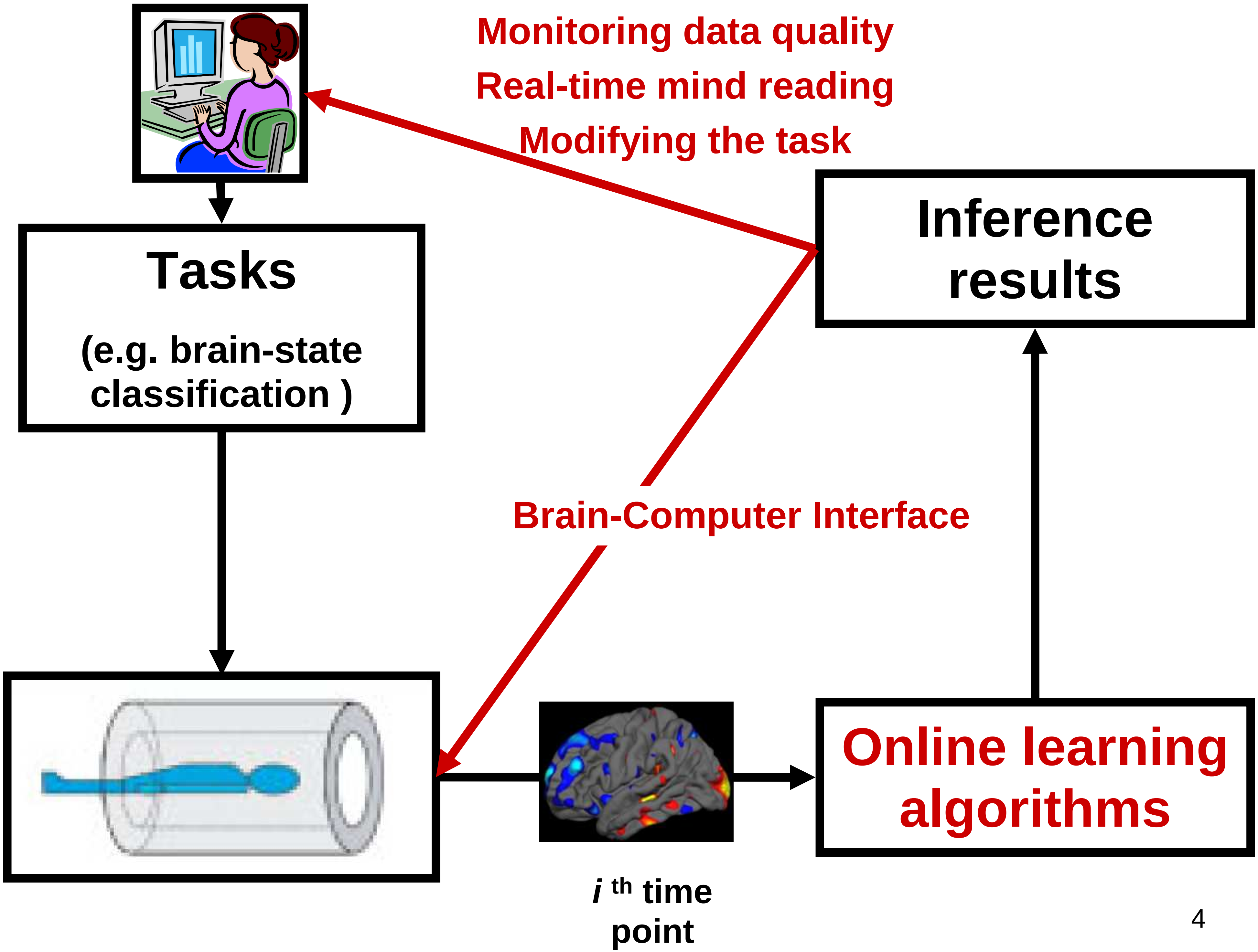
Hao Xu, Yongxin Taylor Xi,
Ray Lee, Peter J. Ramadge

ICASSP 2011

Roadmap

- Real-time fMRI System
- Online Learning algorithms
  - Previous work
  - Real-time Conjugate Gradients (rtCG)
- Test results


Tasks
(e.g. brain-state classification)
Inference results
Offline learning algorithms
time series


Monitoring data quality
Real-time mind reading
Modifying the task
Inference results
Tasks
(e.g. brain-state classification)
Brain-Computer Interface
Online learning algorithms
i th time point

Real-time fMRI System Challenge

- Computational complexity
  - fMRI data (122,880~163,840 voxels per TR)
  - TR = 2-3 secs
- Choice of algorithm
  - Accurate
  - Adaptive

Real-time Conjugate Gradients (rtCG)

- Motivated by Partial Least Squares (PLS)
- Online
  - Both train and classify in real-time
- Fast
  - 0.5 secs in real fMRI tests

    [Intel Dual Core PC 1.83 GHZ CPU, 2GB RAM]
- Competitive accuracy
  - About 90% in real fMRI tests
- Adaptive

Online Learning Algorithms for fMRI

- There are many online learning algorithms. Some have been applied to fMRI. Here are some examples :
- General linear model (GLM)
  - Bagarinao E., Matsuo K., Nakai T., "Real-Time Functional MRI Using a PC Cluster," Concepts. Magn. Reson. Part B, vol. 19B, pp. 14–25, 2003
- Independent component analysis (ICA)
  - Esposito F., Seifritz E., Formisano E. et al., "Real-time independent component analysis of fMRI time-series," NeuroImage, vol. 20, pp. 2209–2224, 2003.
- Support vector machine (SVM)
  - Yongxin Taylor Xi, Hao Xu, Ray Lee, Peter J. Ramadge, "Online Kernel SVM for real-time fMRI brain state prediction," ICASSP 2011.

Partial Least Squares (PLS)

[Wold *et al.* 1984]

Input $X = [x_1, \ldots, x_n]^T \in R^{n \times k}$ Output $y = [l_1, \ldots, l_n]^T \in R^n$

$$X = \sum_{i=1}^{m} f_i p_i^T + D = FP^T + D$$

$$y = \sum_{i=1}^{m} f_i q_i + e = Fq + e$$

latent factors $f_i \in R^n, i = 1, \ldots, m$

$$F = [f_1, \ldots, f_m] \in R^{n \times m}$$

$$P = [p_1, \ldots, p_m] \in R^{k \times m}$$

$$q = [q_1, \ldots, q_m]^T \in R^m$$

Incremental Sparse Bridge PLS (iSB-PLS)

[McWilliams *et al.* 2009]

- Ridge parameter
  - Adding ridge parameter to covariance matrix
- Improvement
  - Extract all latent factors in one step

PLS to rtCG

- Comparable performance
  - PLS is a conjugate gradient algorithm
    - [Aloke Phatak and Frank de Hoog, 2002]
    - [Magnus R. Hestenes and Eduard Stiefel,1952]
- More efficient
  - Real-time conjugate gradient (rtCG)

Online learning algorithm

- Receive new example $x_t \in R^k$ at time t and classifier b_{t-1} makes prediction $\hat{\ell}_t = sign(x_t{}^T b_{t-1})$
- Receive true label $\ell_t \in \{-1,1\}$

  Update classifier $b_{t-1} \Rightarrow b_t = \arg\min_{b \in R^k} \left\| X_t b - y_t \right\|_2^2$

Window parameter h $X_t^h = [x_{t-h+1}, \ldots, x_t]$ $y_t^h = [l_{t-h+1}, \ldots, l_t]$

Efficient Adaptive

$$b_t = \arg\min_{b \in R^k} \left\| X_t^h b - y_t^h \right\|_2^2$$

Conjugate Gradient can solve

Conjugate Gradient (CG)

[Hestenes *et al.* 1984]

- Solve quadratic minimization

$$b_t = \arg\min_{b \in R^k} \left\| X_t^h b - y_t^h \right\|_2^2$$

- CG is an iterative metho
  - Start at $\boxed{\mathrm{b}^0 = \vec{0}}$
  - Search directions d_i are

    $d_i^T (X_t^h)^T X_t^h d_j = 0 \quad i \neq j$
  - Line search on each direction
  - Terminate in $\boxed{h}$ steps $\mathrm{b}^h \to b_t$

$X_{t-1}^h \to X_t^h \quad \mathrm{b}^0 = b_{t-1}$

Terminate in $I_{\max}$

Past memory vs.

learning from recent data

Comparison with PLS

- Theorem 1

  Assuming that $b_0 = 0$, rtCG and PLS yield identical steps when applied to X_t^h and y_t^h.

  rtCG $\Leftrightarrow$ PLS under certain initializations

  Comparable performance

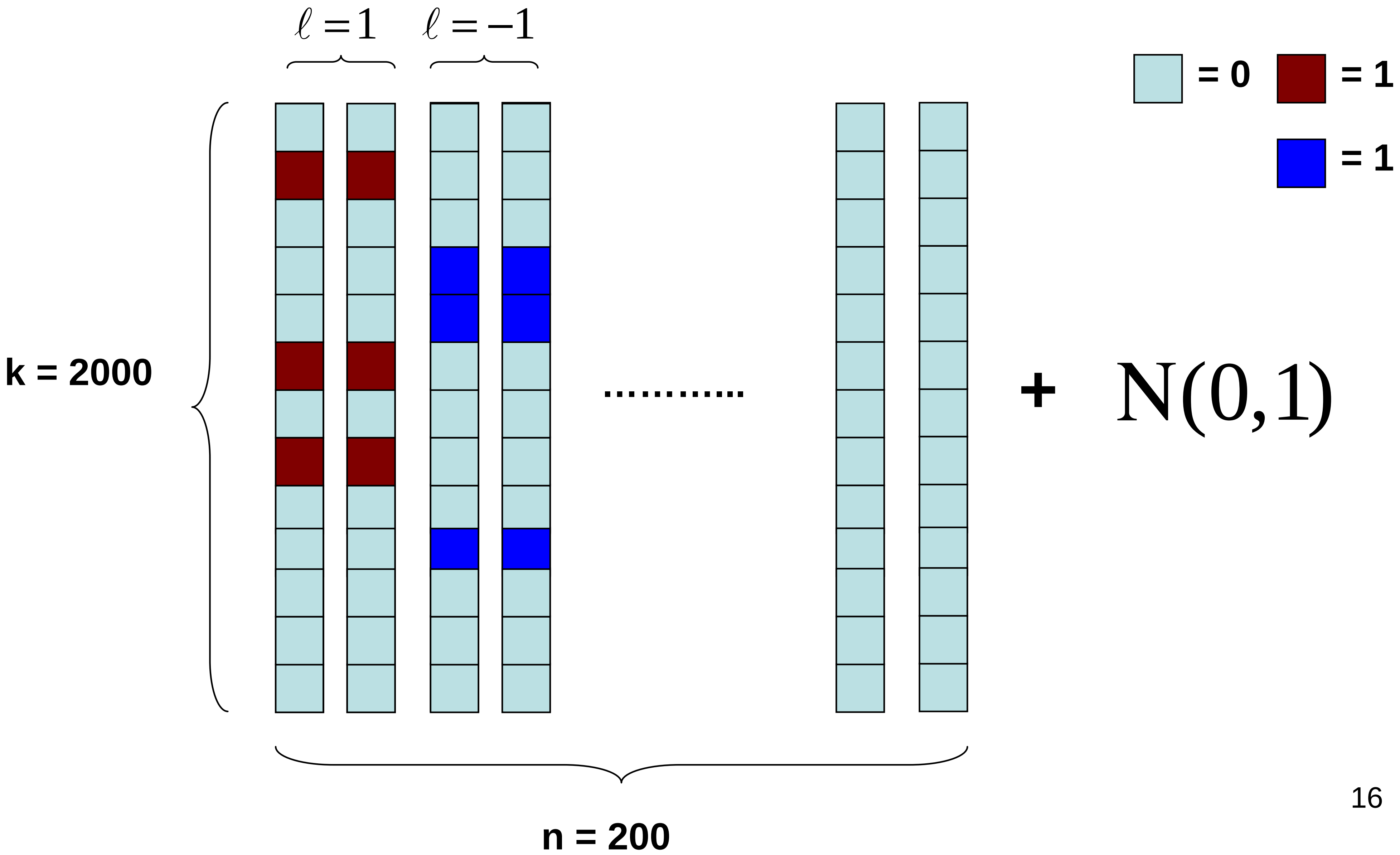
Roadmap

- Real-time fMRI System
- Online Learning algorithms
  - Previous work
  - Real-time Conjugate Gradients (rtCG)
- Test results

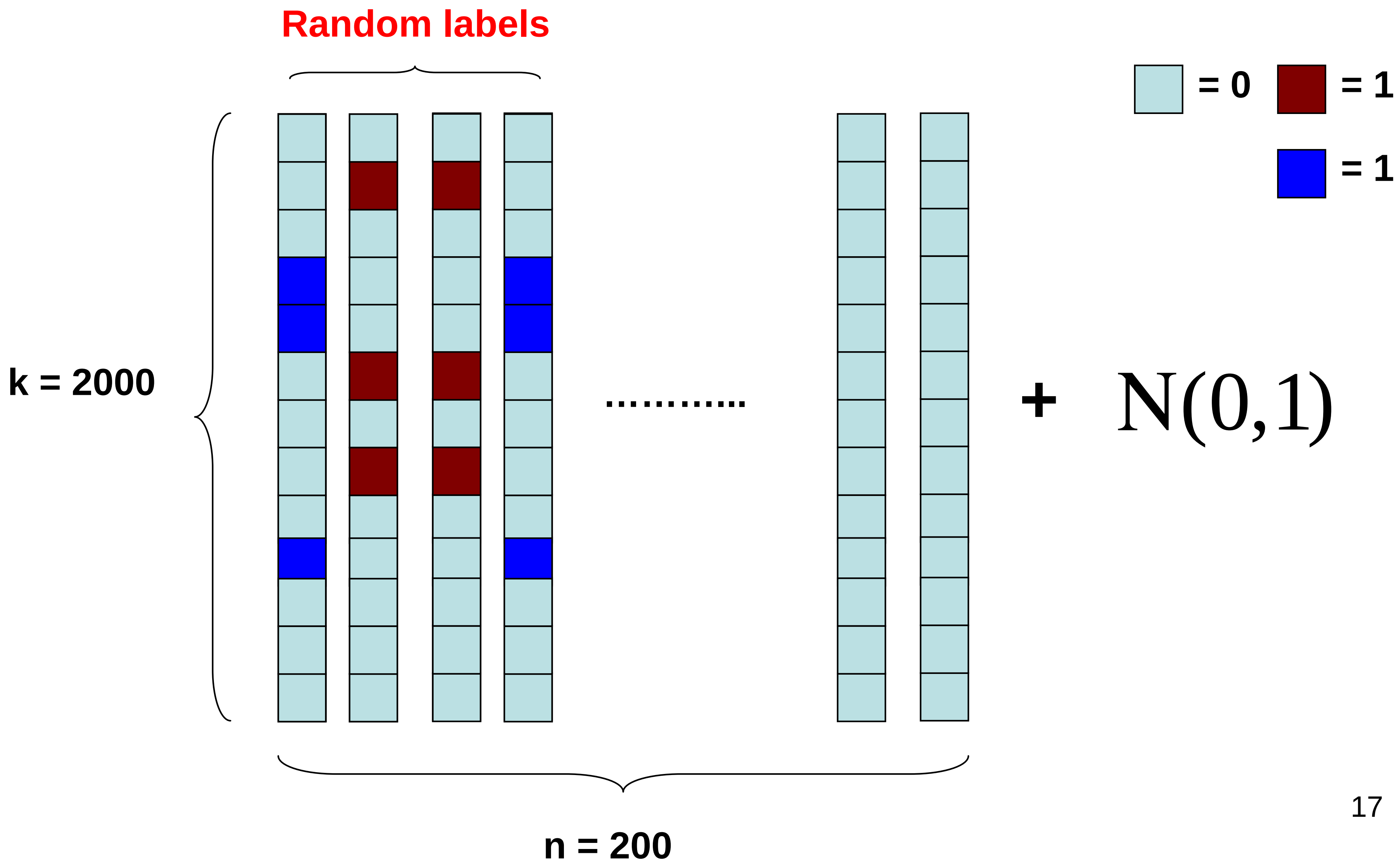
Test algorithms

- Real-time CG (rtCG)
- Real-time PLS (rtPLS)
  - PLS applied to X_t^h and y_t^h
- Real-time BPLS (rtBPLS)
  - iSB-PLS applied to X_t^h and y_t^h , with its sparse setting part removed

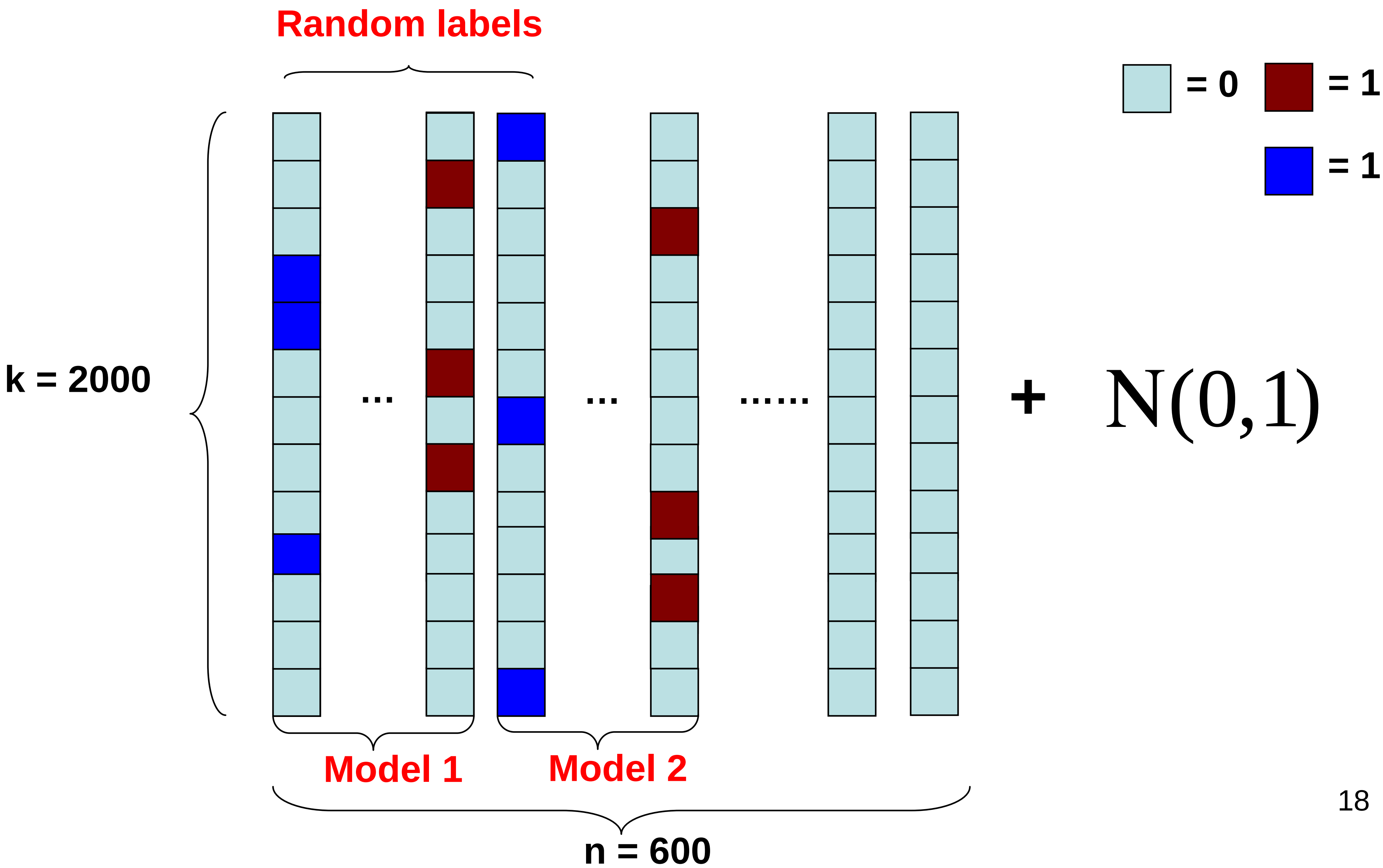
Synthetic Data Test 1



Synthetic Data Test 2

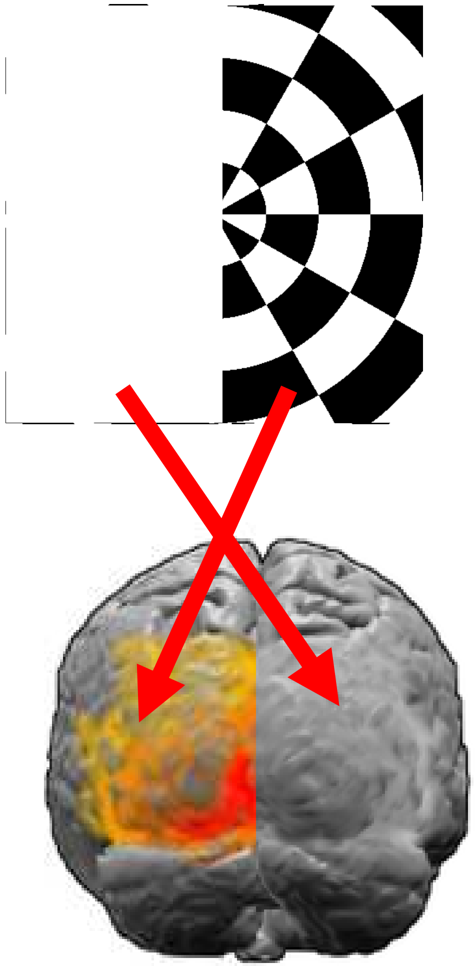


Synthetic Data Test 3



fMRI Data Test 1

- Classification task
  - Position of object: left/right
- Stimulus:
  - left/right checker board
  - interval: 15s
- Input
  - fMRI image every 3 seconds
  - Dimension = 122,880



fMRI Data Test 2

- Classification task
  - Position of object: random

fMRI Data Test 3

- Classification task
  - category-related object vision **[Haxby *et al.* 2001]**
- Data:
  - 10 runs of one subject
  - Faces vs. Houses

[Princeton Neuroscience Institute, “MVPA Toolbox”]

- Input
  - Dimension = 163,480

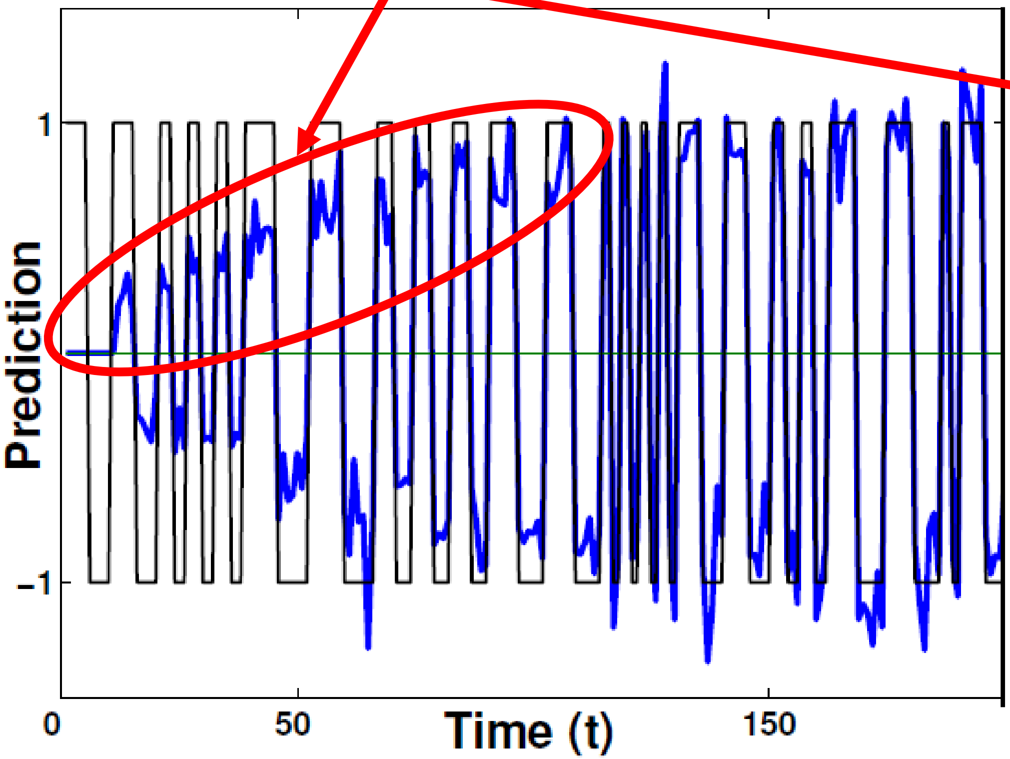
Test results

	rtCG	rtPLS	rtBPLS
	accuracy (%) / average time (ms)		
Synth 1	**100** / **2.3**	**100** / 10.3	**100** / 7.0
Synth 2	**100** / **2.3**	**100** / 10	**100** / 6.9
Synth 3	97.97 / **2.4**	98.81 / 9.9	**99.15** / 7.1
fMRI 1	**92.67** / **414.9**	74 / 1490.3	73.33 / 867.5
fMRI 2	**91.05** / **408.2**	75.26 / 1472.7	73.16 / 904.5
fMRI 3	**90.23** / **548.1**	89.55 / 2129.1	88.06 / 1354.8

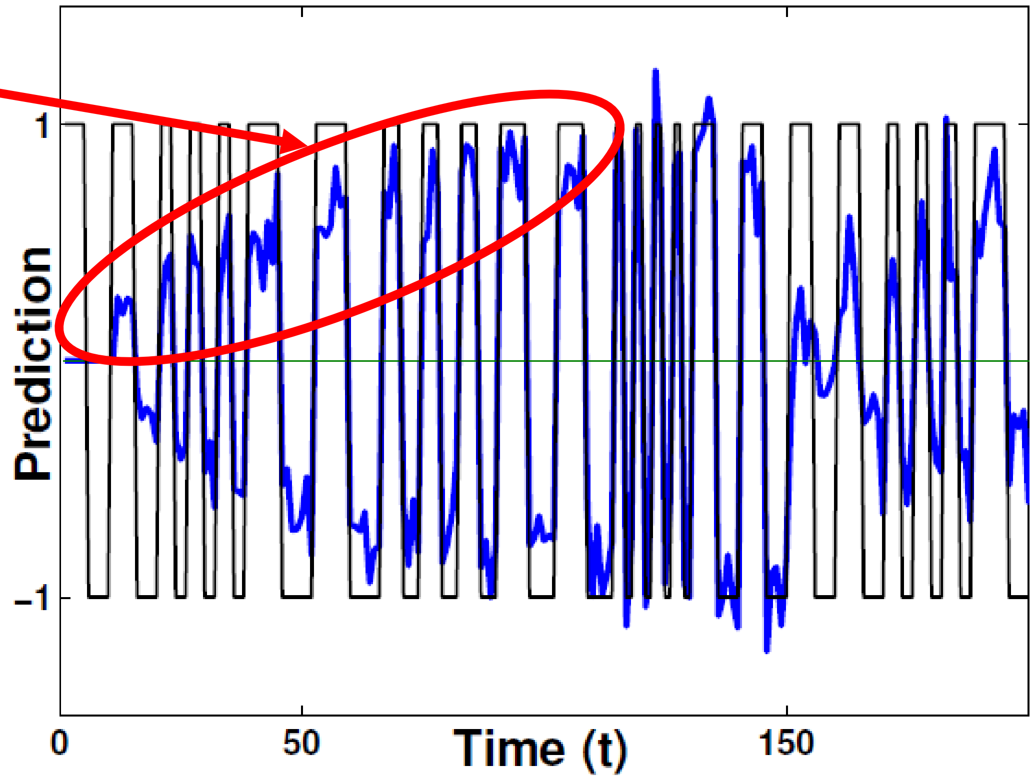
Same number of iterations were used in three algorithms.

$\mathrm{b}^0 = b_{t-1}$

Encode memory



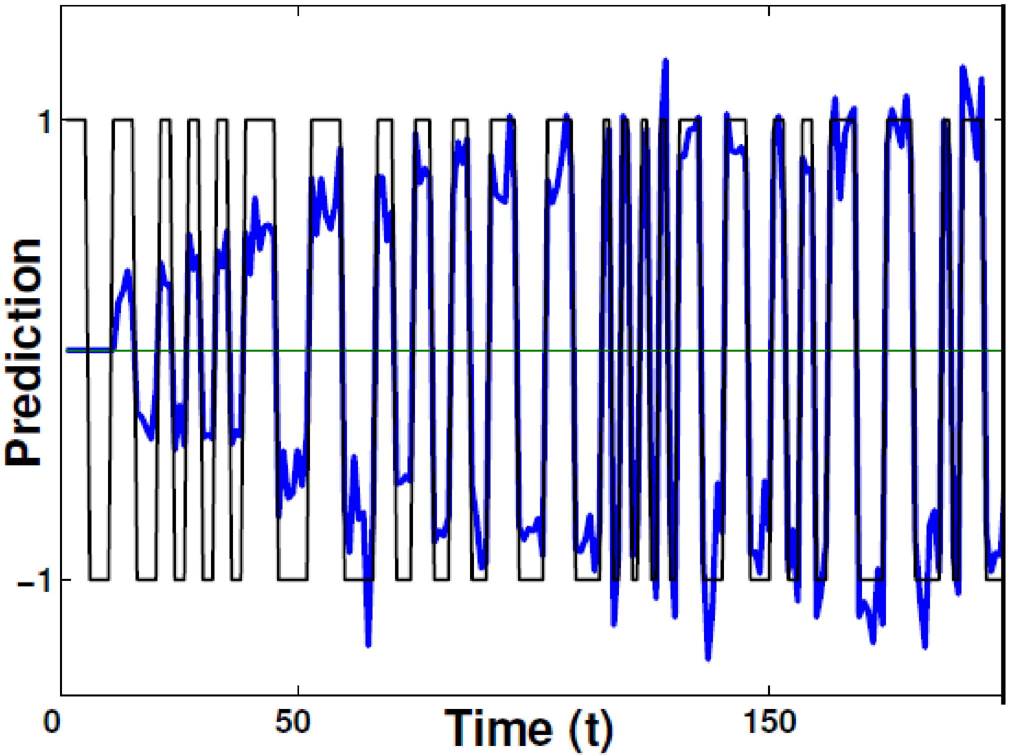
Prediction (Synthetic 2)



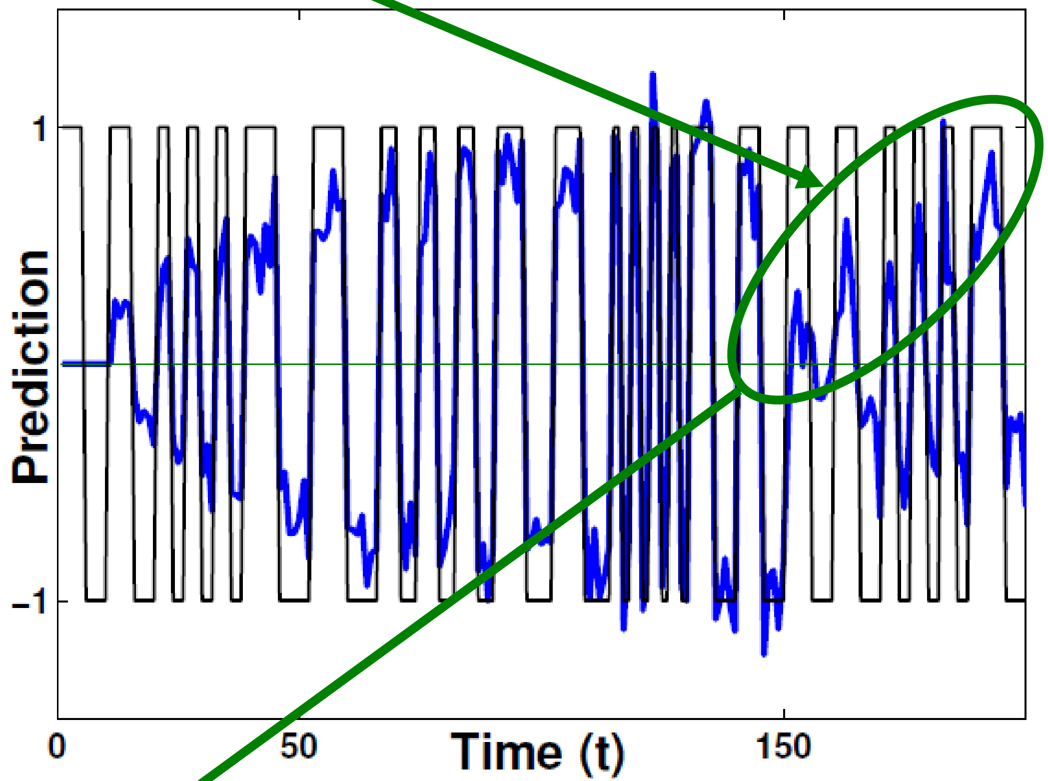
Prediction (Synthetic 3)

$\mathrm{b}^0 = b_{t-1}$

Memory vs. Adaptiveness

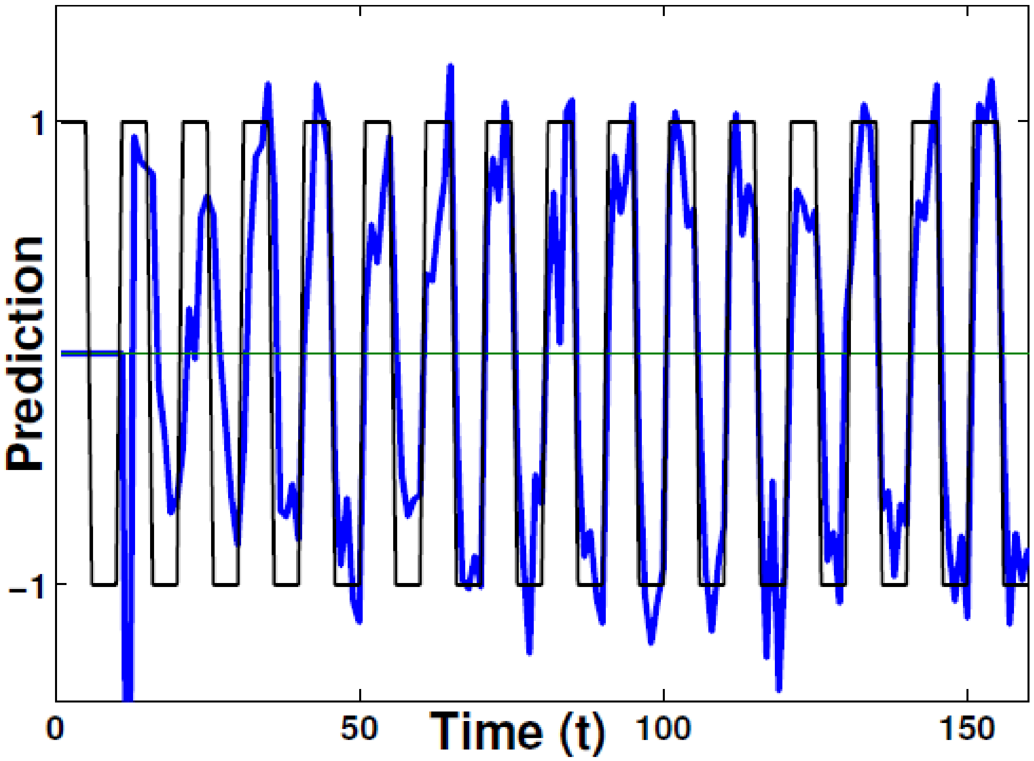


Prediction (Synthetic 2)

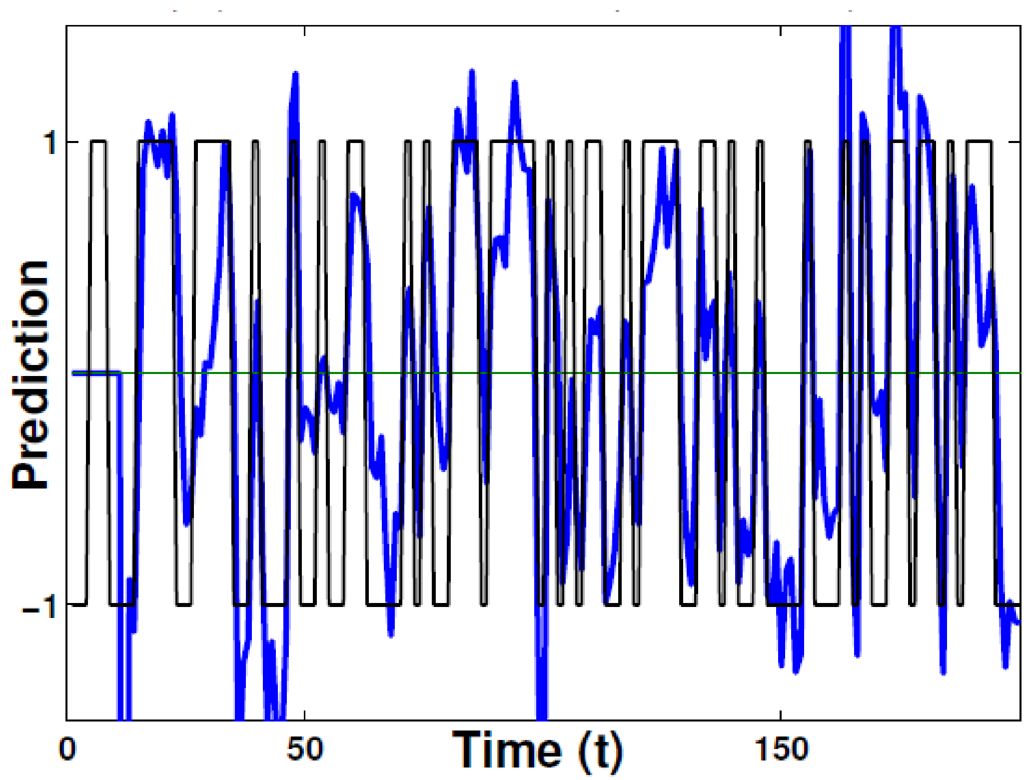


Prediction (Synthetic 3)

	rtCG	rtPLS	rtBPLS
	accuracy (%) / average time (ms)		
Synth 3	97.97 / 2.4	98.81 / 9.9	**99.15** / 7.1



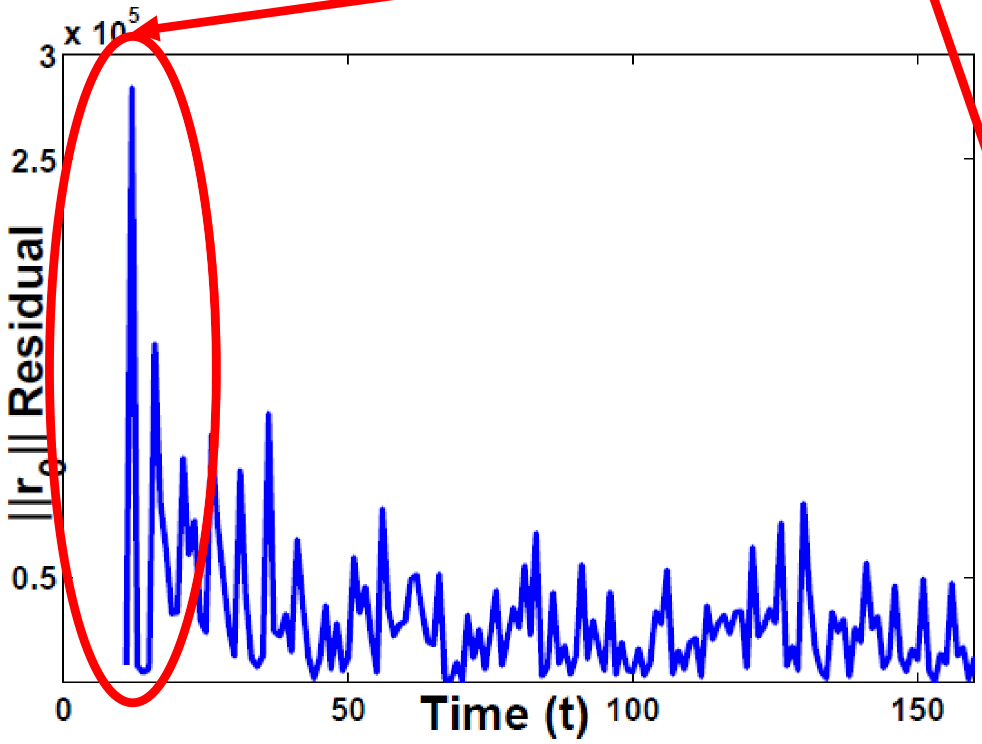
Prediction (fMRI 1)



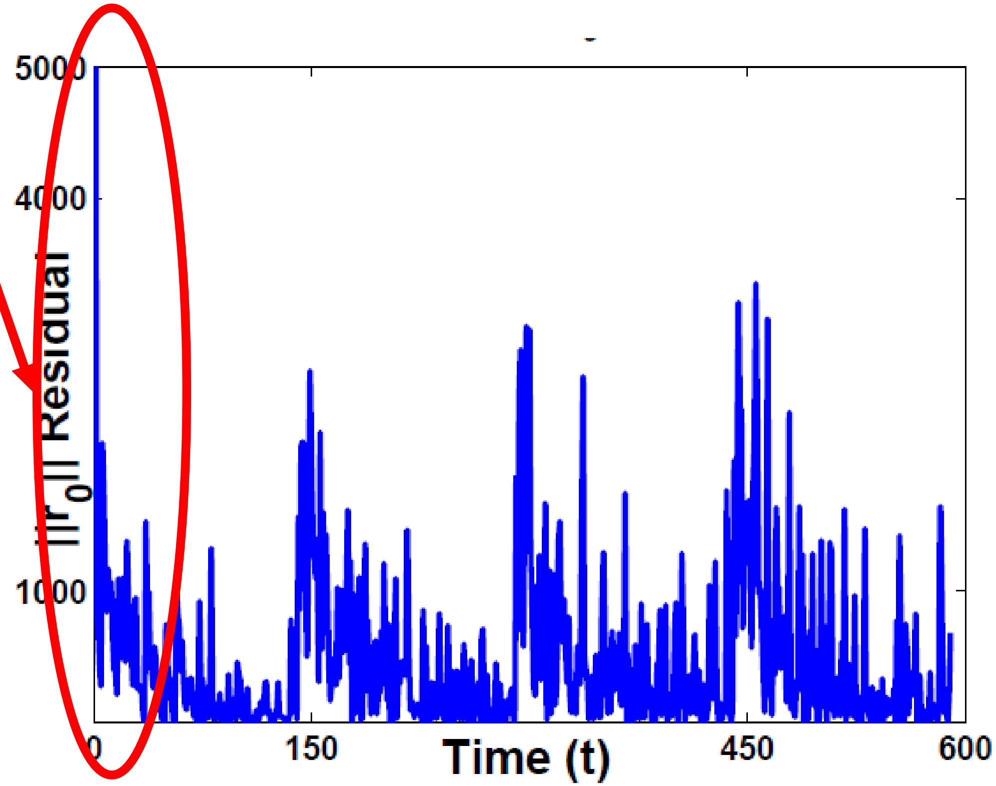
Prediction (fMRI 2)

$b^0 = b_{t-1}$

Faster

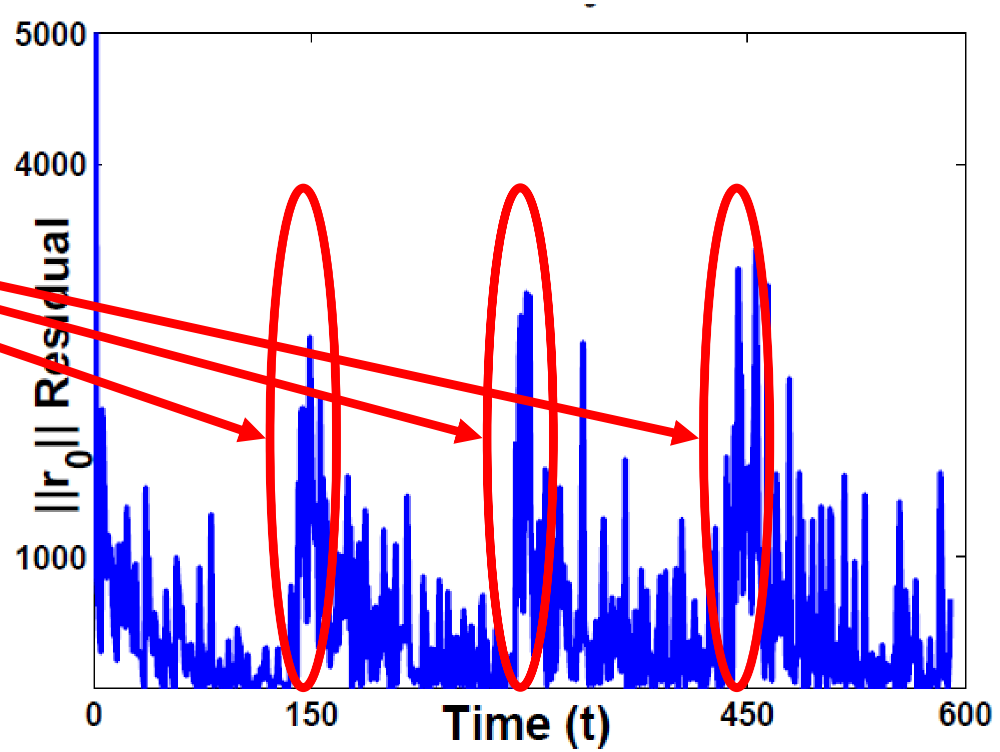


Residual (fMRI 1)



Residual (Synthetic 3)

Model changes
Start over



Residual (Synthetic 3)

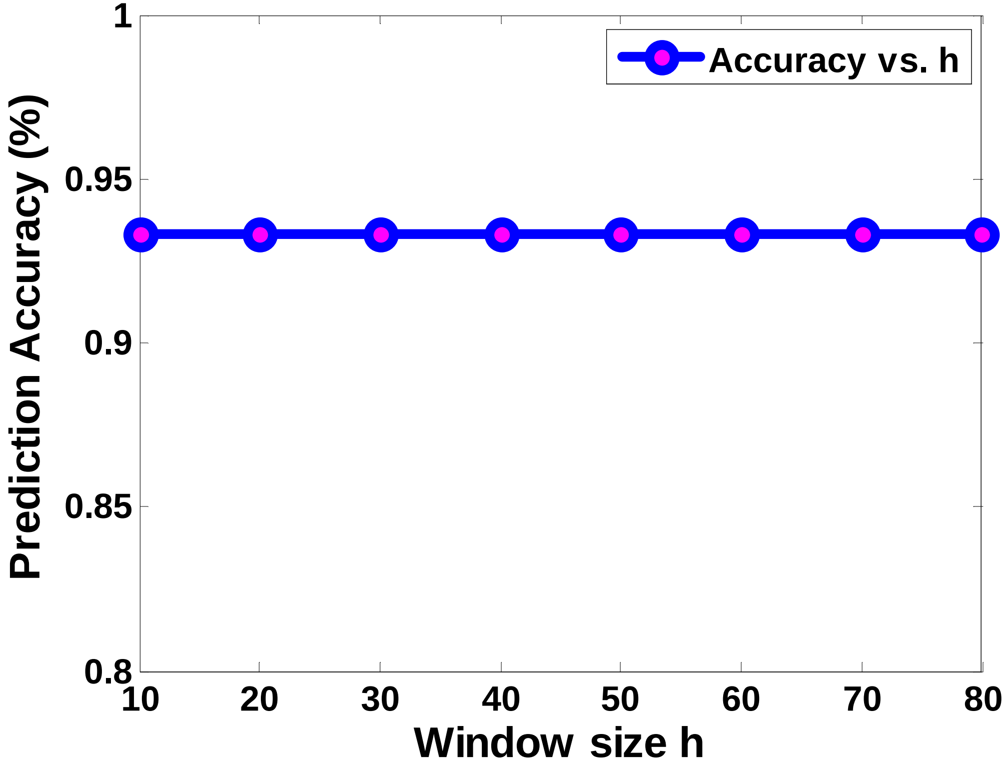
99.15 / 2.5

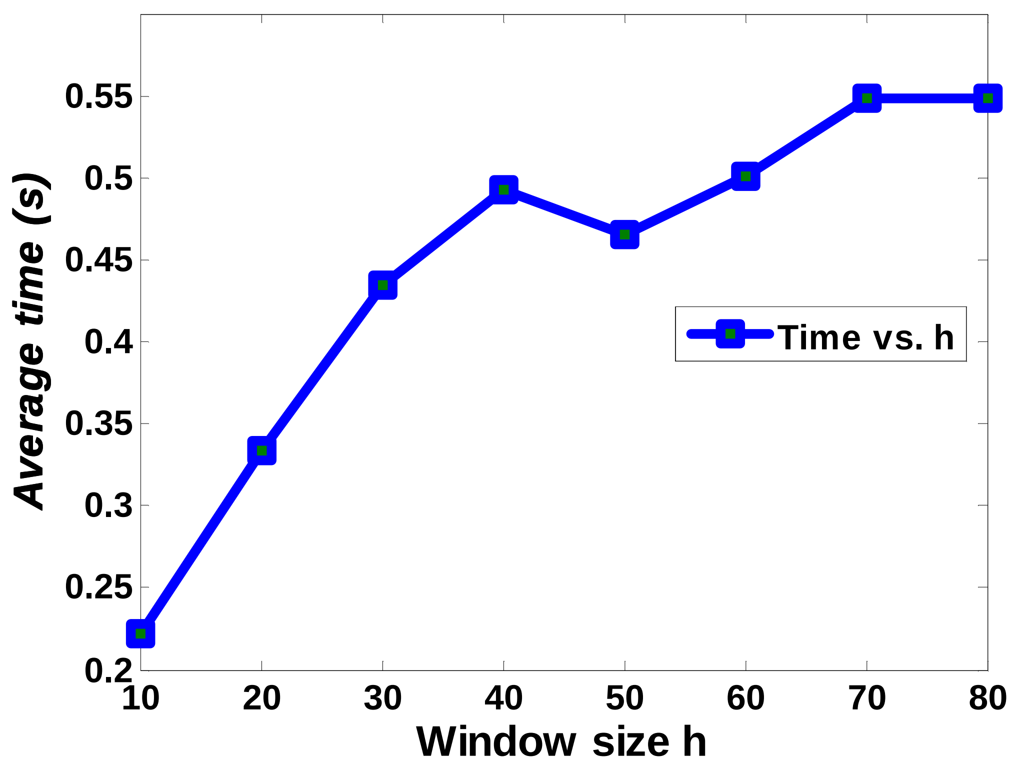
	rtCG	rtPLS	rtBPLS
	accuracy (%) / average time (ms)		
Synth 3	97.97 / 2.4	98.81 / 9.9	99.15 / 7.1

Future work

- Using prediction results
- More complicated experiments
- Comparison with other real-time algorithms
  - ICASSP 2011 Poster 9 in session MLSP-P1
    Online Kernel SVM for Real-time fMRI Brain State Prediction

Thank you!

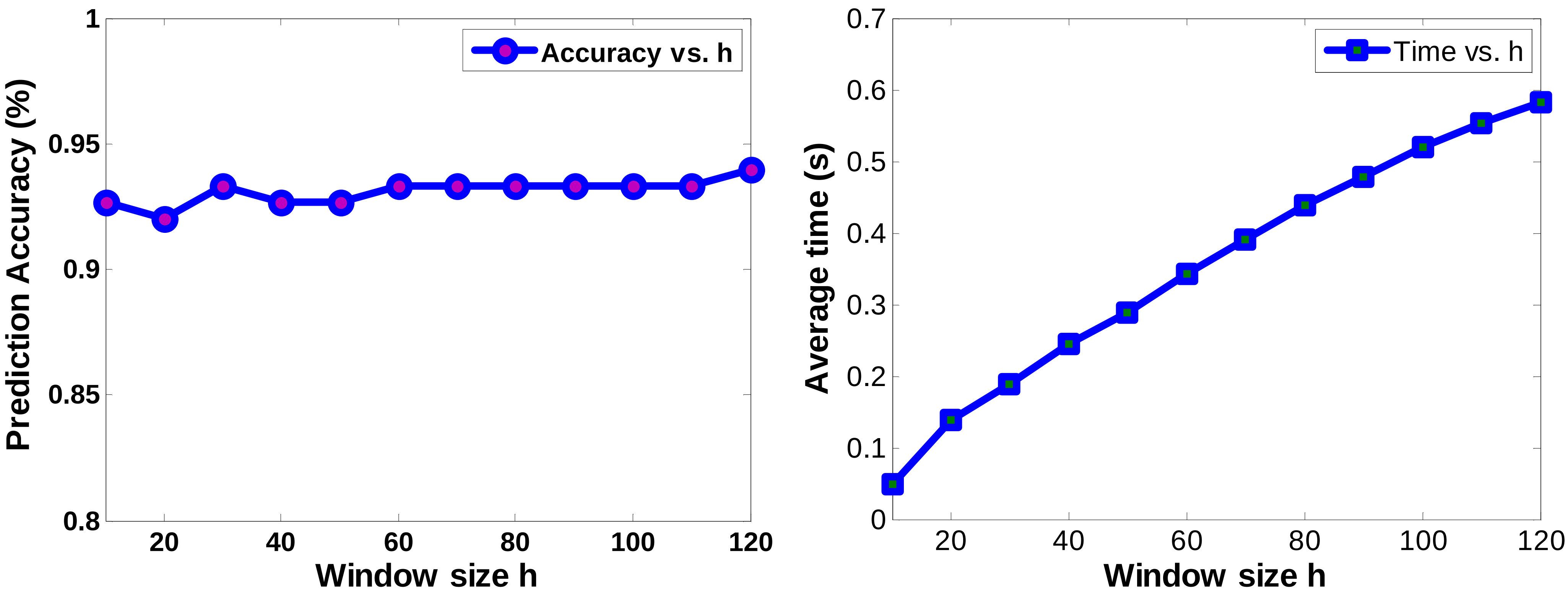




$$I_{\max} = 10 \sim 80$$

$$h = 80$$

$$\text{error tolerance} = 10^{-10}$$



$$I_{\max} = h$$

$$h = 10 \sim 120$$

$$\text{error tolerance} = 10^{-10}$$

rtCG vs. OKSVM & rtLIBSVM

	rtCG	OKSVM	rtLIBSVM
Synth 1	**100** / **1.6**	99.44 / 1.9	97.37 / 4.6
Synth 2	**100** / **2.0**	**100** / **2.0**	97.37 / 4.6
Synth 3	**95.59** / **1.8**	93.26 / 2.0	94.3 / 4.8
fMRI 1	**94.96** / 76	93.53 / **41.2**	65.33 / 401.9
fMRI 2	**90. 5** / 73.3	83.24 / **31.1**	63.16 / 419.4
fMRI 3	**90.62** / 114.4	74.21 / **34.8**	87.65 / 526.5