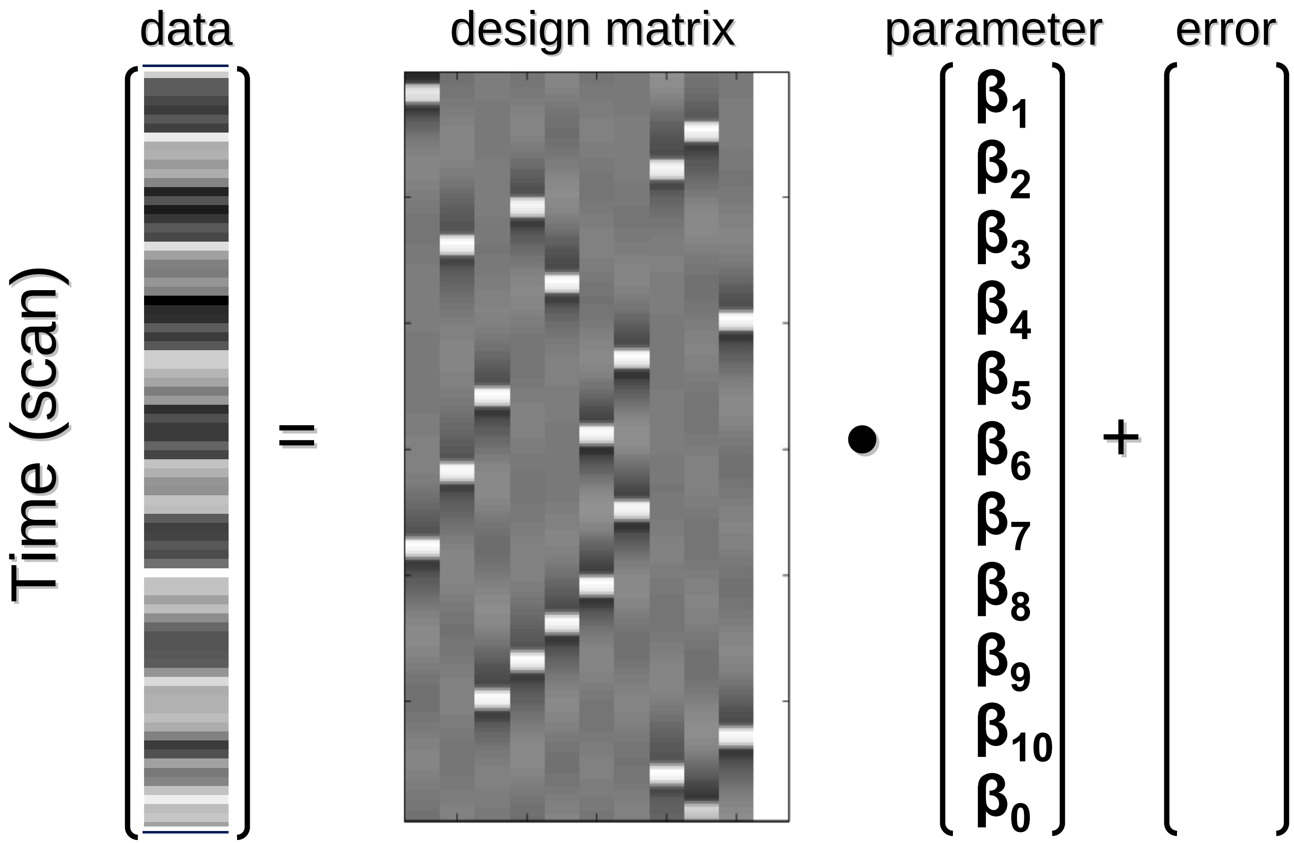
Multivariate Pattern Classification

Thomas Wolbers
Space and Aging Laboratory
Centre for Cognitive and Neural Systems

Outline

- **WHY PATTERN CLASSIFICATION?**
- PROCESSING STREAM
- PREPROCESSING / FEATURE REDUCTION
- CLASSIFICATION
- EVALUATING RESULTS
- APPLICATIONS

Why pattern class.?



data

design matrix

parameter

error

Time (scan)

$$= \bullet \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \beta_4 \\ \beta_5 \\ \beta_6 \\ \beta_7 \\ \beta_8 \\ \beta_9 \\ \beta_{10} \\ \beta_0 \end{bmatrix} +$$

$$y = X \bullet \beta + \varepsilon$$

GLM: separate model fitting for each voxel

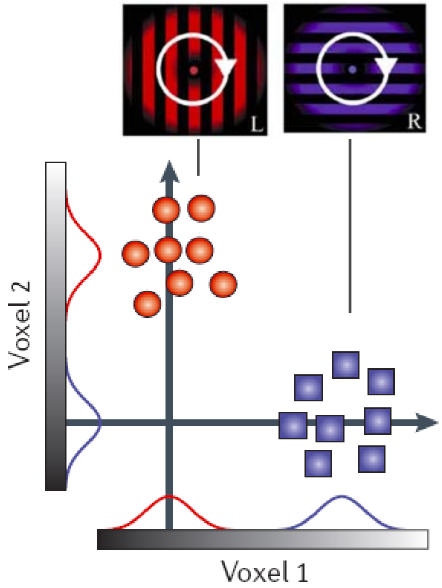
➔ mass-univariate analysis!

Why pattern class.?

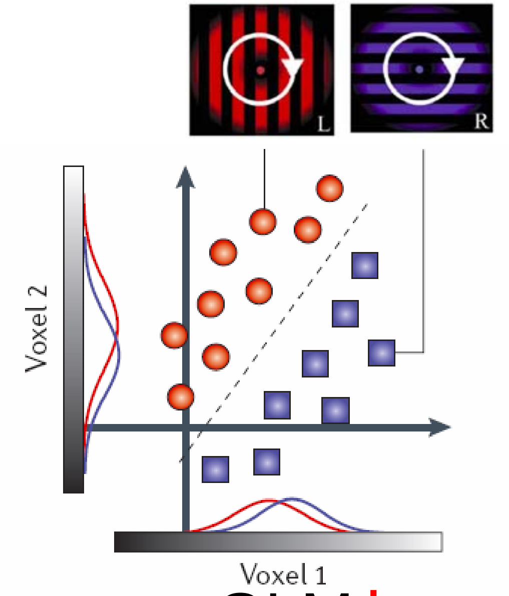
Key idea behind pattern classification

- GLM analysis relies exclusively on the information contained in the time course of **individual** voxels
- Multivariate analyses take advantage of the information contained in activity patterns across space, from multiple voxels

➔ Cognitive/Sensorimotor states are expressed in the brain as ***distributed patterns of brain activity***



GLM ↑



GLM ↓

Why pattern class.?

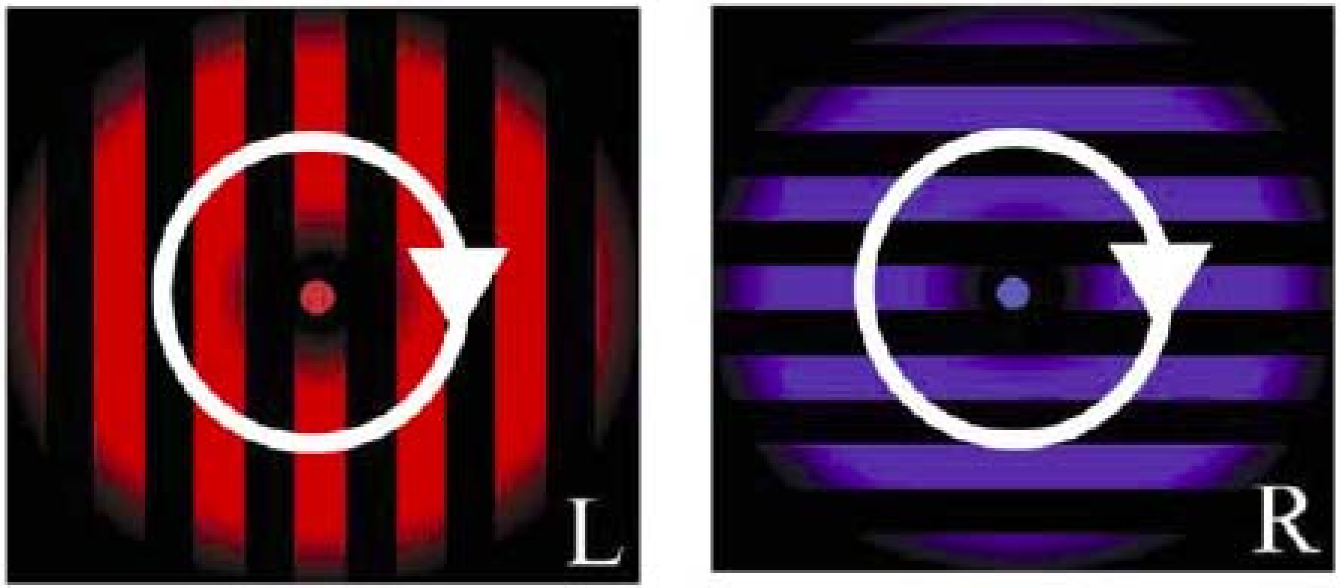
Advantages of multivariate pattern classification

- increase in sensitivity: weak information in single voxels is accumulated across many voxels
- multiple regions/voxels may only carry info about brain states when jointly analyzed
- can prevent information loss due to spatial smoothing (but see Op de Beeck, 2009 / Kamitani & Sawahata 2010)
- can preserve temporal resolution instead of characterizing average response across many trials

Outline

- ➢ WHY PATTERN CLASSIFICATION?
- ➢ **PROCESSING STREAM**
- ➢ PREPROCESSING / FEATURE REDUCTION
- ➢ CLASSIFICATION
- ➢ EVALUATING RESULTS
- ➢ APPLICATIONS

BINOCULAR RIVALRY

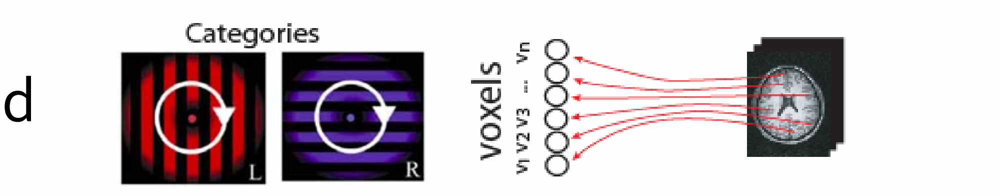


Can spontaneous changes in conscious experience be decoded from fMRI signals in early visual cortex?

Haynes & Rees (2005). Current Biology

Processing stream

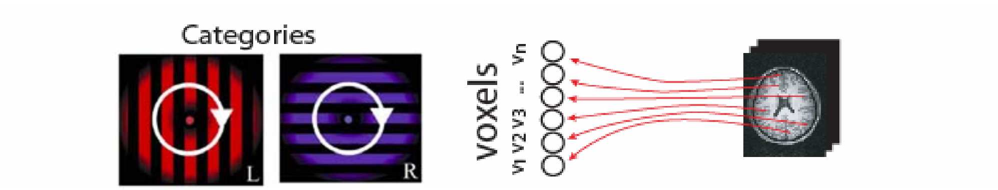
1. Acquire fMRI data while subject is viewing blue and red gratings



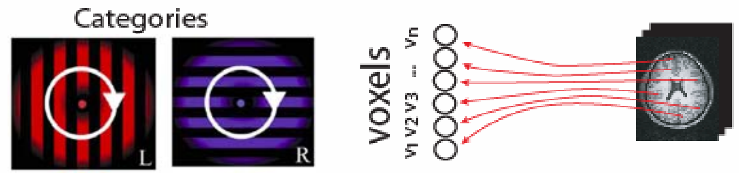
Processing stream

1. Acquire fMRI data

2. Preprocess fMRI data



Processing stream



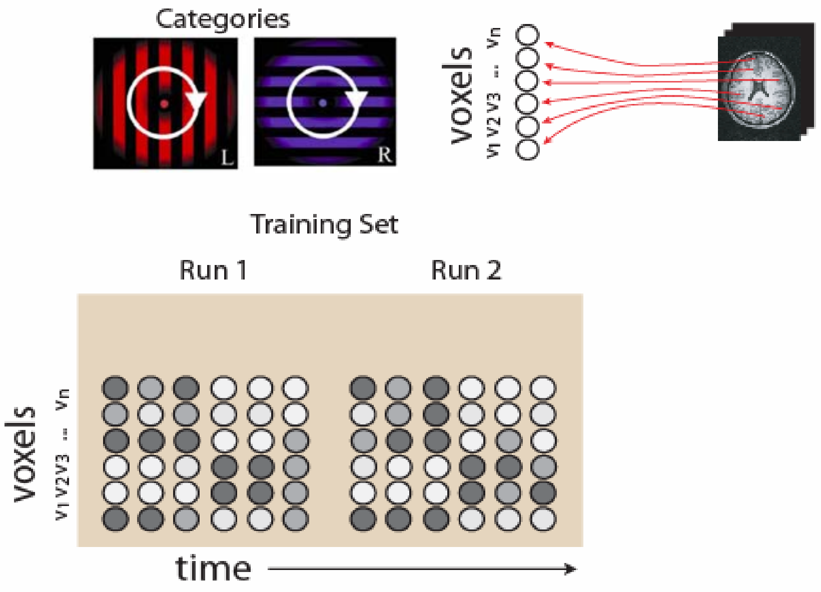
1. Acquire fMRI data
2. Preprocess fMRI data

3. Select relevant features (i.e. voxels)

Processing stream

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features

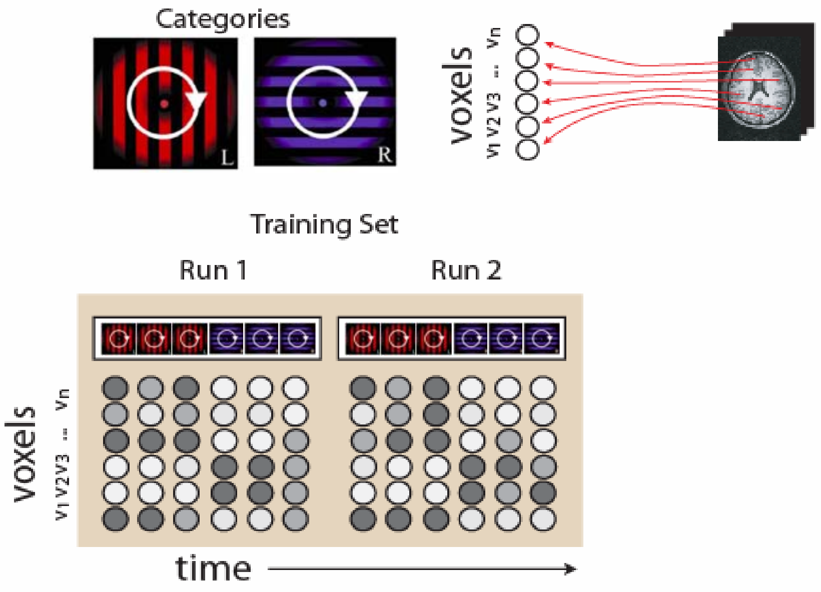
4. Convert each fMRI volume into a vector that reflects the *pattern of activity across voxels* at that point in time.



Processing stream

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features
4. Generate fMRI patterns

5. Label fMRI patterns according to whether the subject was perceiving blue vs. red (adjusting for hemodynamic lag)

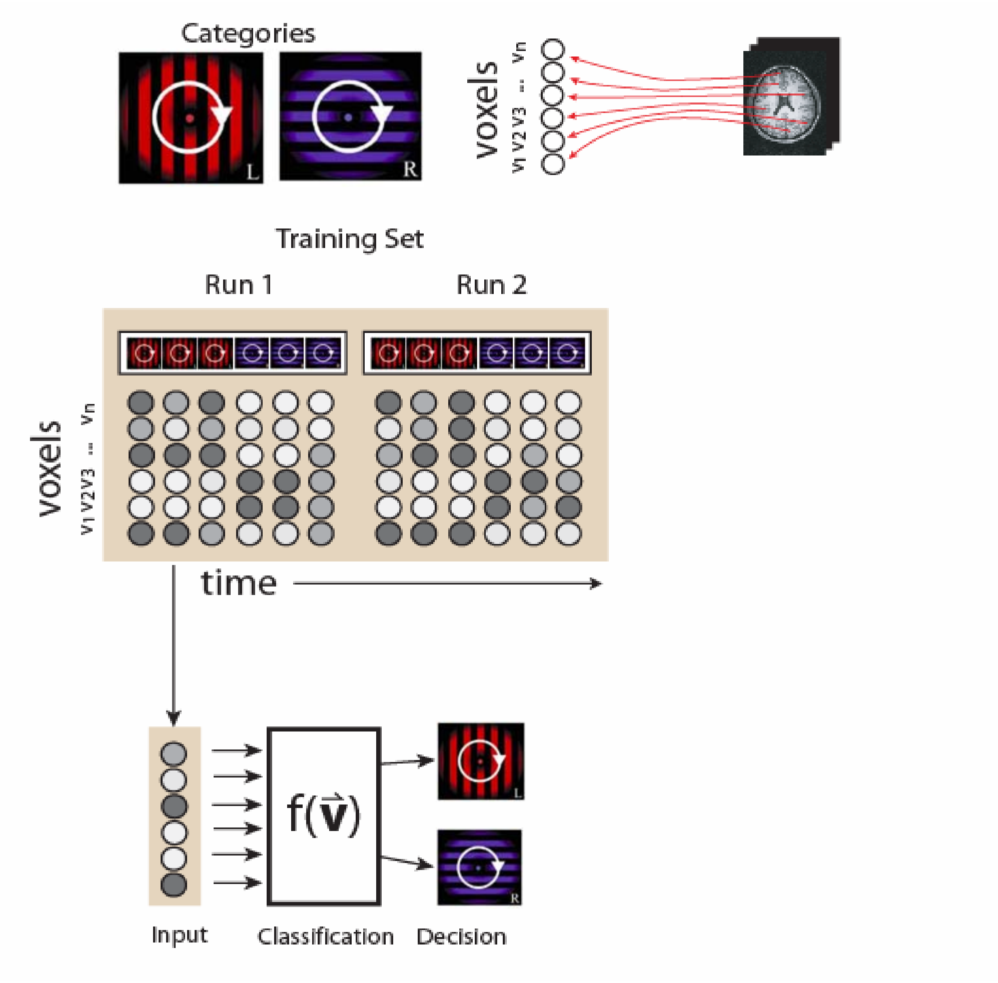


Processing stream

SPM Course Edinburgh 2010

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features
4. Generate fMRI patterns
5. Label fMRI patterns

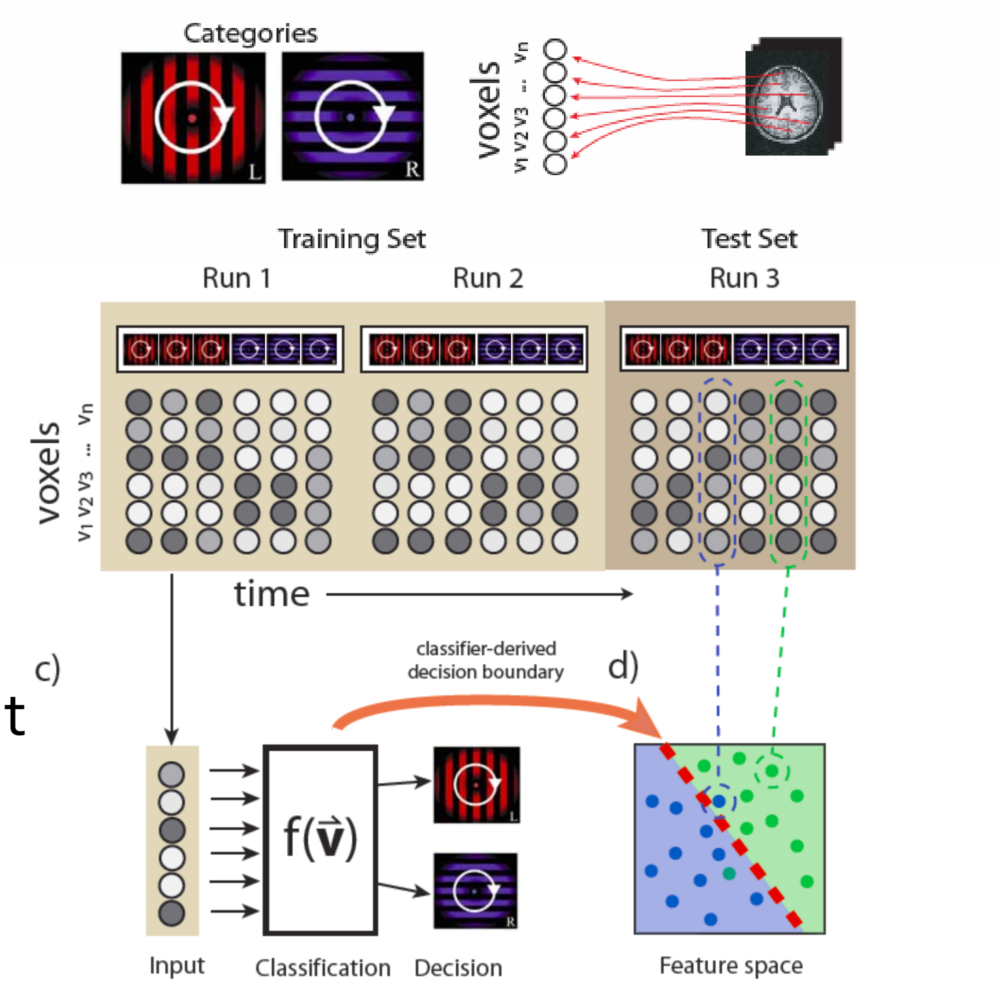
6. Train a classifier to discriminate between blue patterns and red patterns



Processing stream

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features
4. Generate fMRI patterns
5. Label fMRI patterns
6. Train the classifier

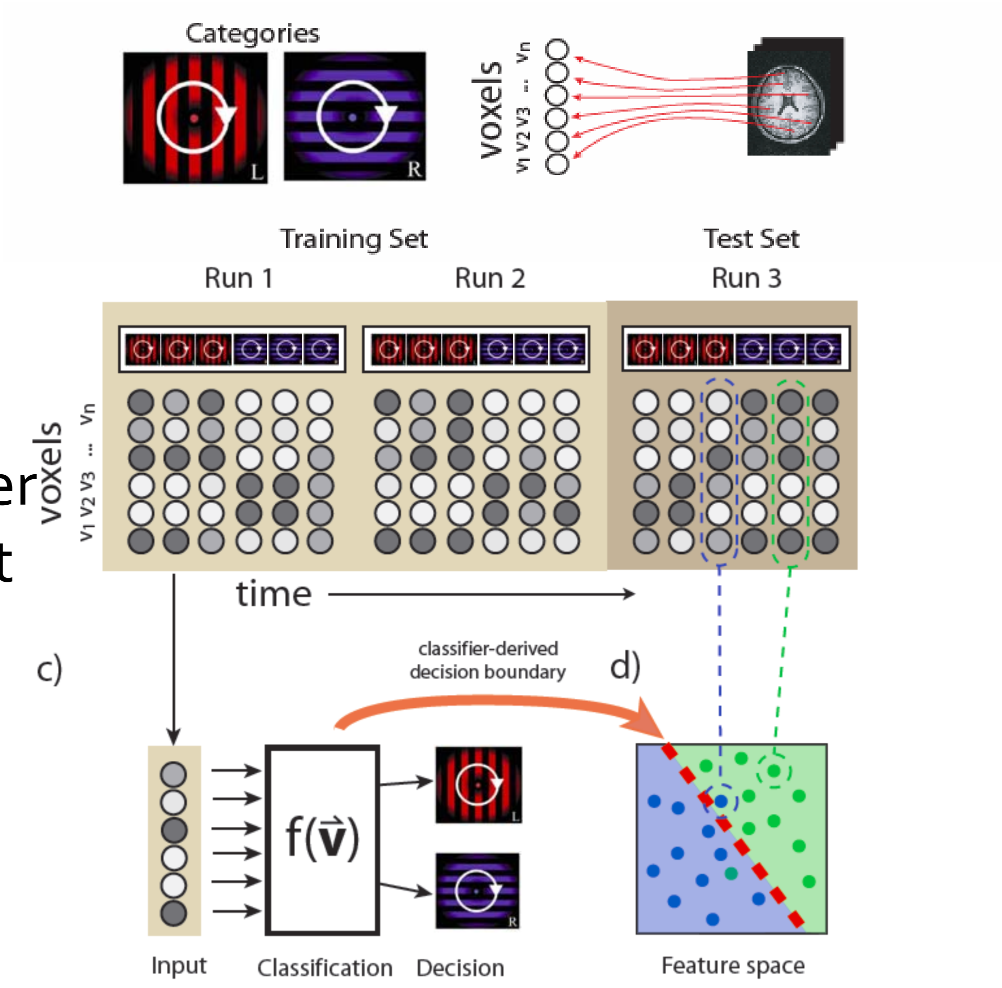
7. Apply the trained classifier to new fMRI patterns (not presented at training).



Processing stream

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features
4. Generate fMRI patterns
5. Label fMRI patterns
6. Train the classifier
7. Apply the trained classifier to new fMRI patterns (not presented at training).

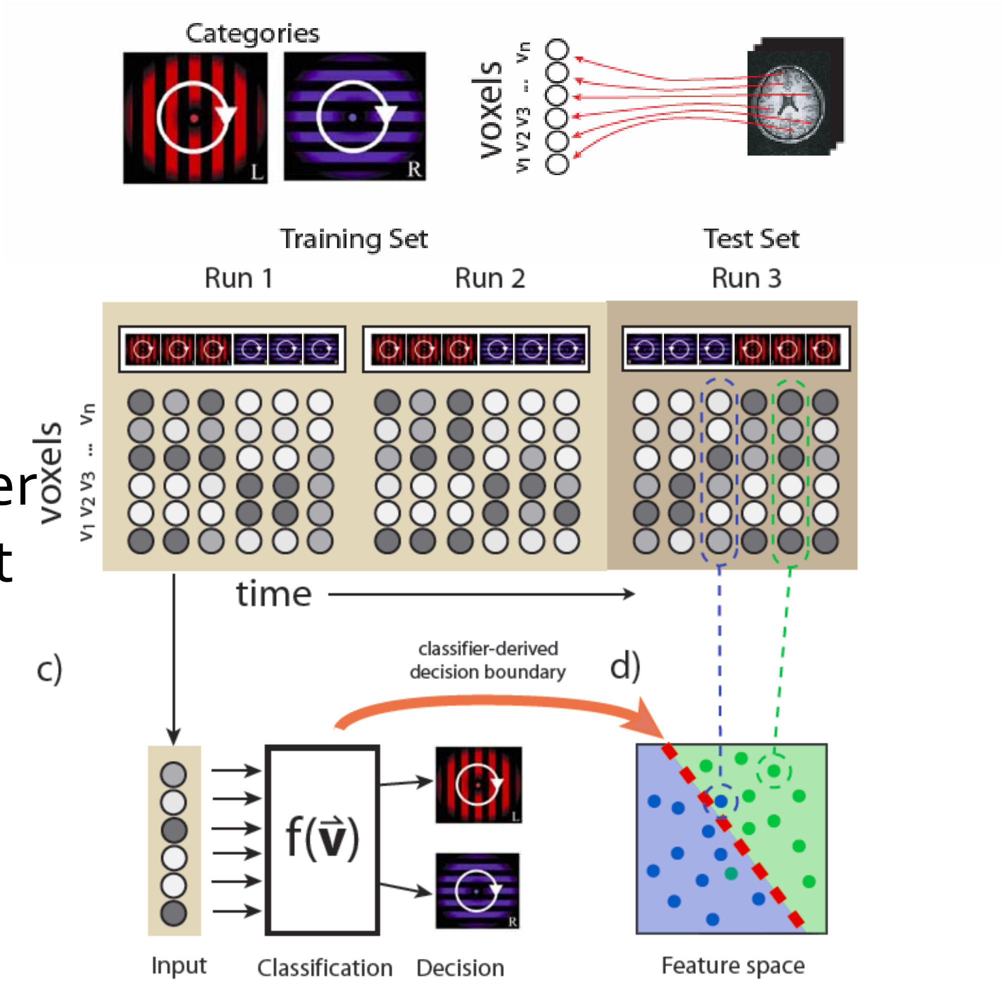
8. Crossvalidation

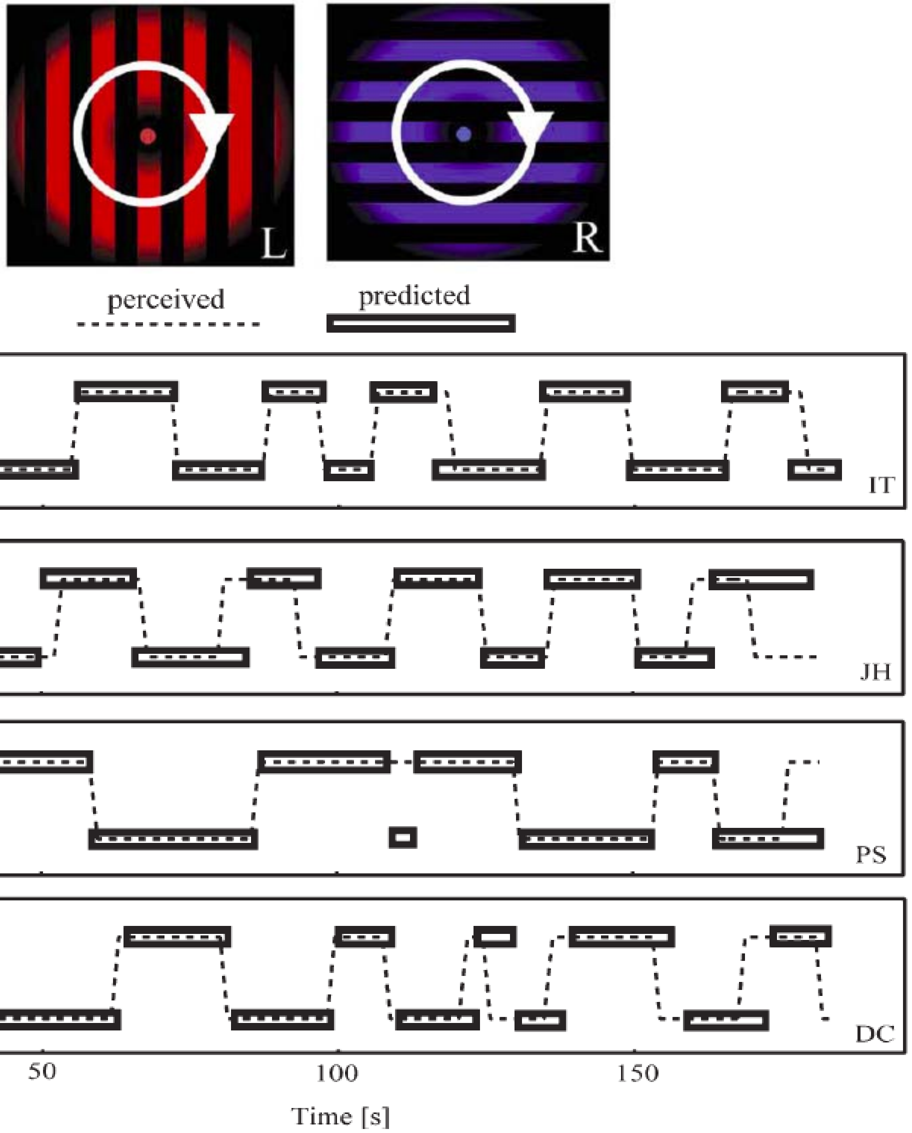


Processing stream

1. Acquire fMRI data
2. Preprocess fMRI data
3. Select features
4. Generate fMRI patterns
5. Label fMRI patterns
6. Train the classifier
7. Apply the trained classifier to new fMRI patterns (not presented at training).
8. Crossvalidation

9. Statistical inference





Haynes & Rees (2005). Current Biology

Outline

- WHY PATTERN CLASSIFICATION?
- PROCESSING STREAM
- **PREPROCESSING / FEATURE REDUCTION**
- CLASSIFICATION
- EVALUATING RESULTS
- APPLICATIONS

Preprocessing

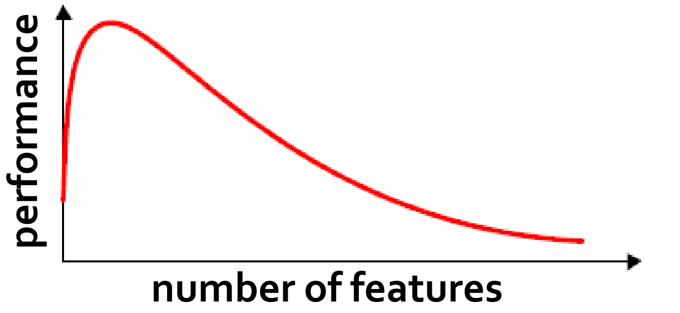
1. **(Slice Timing +) Realignment (SPM, FSL ...)**
2. **High-pass filtering / Detrending**
   - remove linear (and quadratic) trends (i.e. scanner drift)
   - remove low-frequency artifacts (i.e. biosignals)
3. **Z-Scoring**
   - remove baseline shifts between scanning runs
   - reduce impact of outliers

Feature Reduction

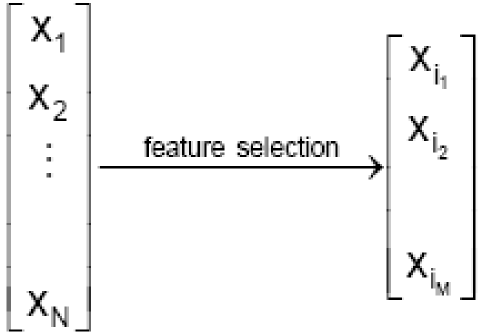
The problem

- fMRI data are typically sparse, high-dimensional and noisy
- Classification is sensitive to information content in all voxels

➔ many uninformative voxels = poor classification (i.e. due to overfitting)



Solution 1: Feature selection



- select subset with the most informative features
- original features remain unchanged

Feature Selection

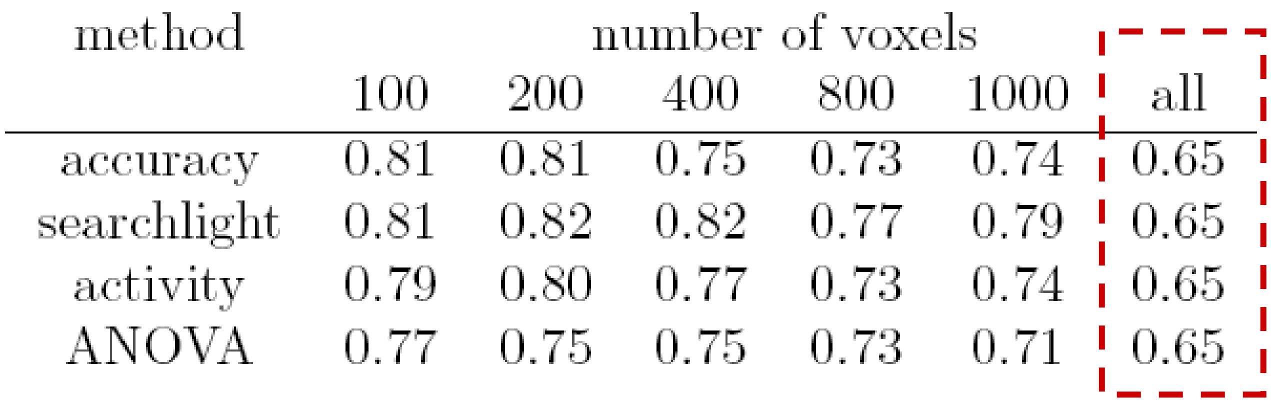
'External' Solutions

- Anatomical regions of interest
- Independent functional localizer (Haynes & Rees: retinotopic mapping to identify early visual areas)
- Searchlight classification: define region of interest (i.e. sphere) and move it across the search volume → exploratory analysis

'Internal' univariate solutions

- activation vs. baseline (t-Test)
- mean difference between conditions (ANOVA)
- single voxel classification accuracy

Feature Selection



method	number of voxels					
	100	200	400	800	1000	all
accuracy	0.81	0.81	0.75	0.73	0.74	0.65
searchlight	0.81	0.82	0.82	0.77	0.79	0.65
activity	0.79	0.80	0.77	0.73	0.74	0.65
ANOVA	0.77	0.75	0.75	0.73	0.71	0.65

Pereira et al. (2009)

Peeking #1 (ANOVA and classification only)

- testing a trained classifier needs to be performed on *independent* test datasets
- if entire dataset is used for feature selection, classification estimates become overly optimistic

➔ **nested cross-validation!**

Feature Extraction

Solution 1: Feature selection

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ x_N \end{bmatrix} \xrightarrow{\text{feature selection}} \begin{bmatrix} x_{i_1} \\ x_{i_2} \\ \\ x_{i_M} \end{bmatrix}$$

- select subset from all available features
- original features remain unchanged

Solution 2: Feature extraction

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ x_N \end{bmatrix} \xrightarrow{\text{feature extraction}} \begin{bmatrix} y_1 \\ y_2 \\ \\ y_M \end{bmatrix} = f\left(\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \\ x_N \end{bmatrix} \right)$$

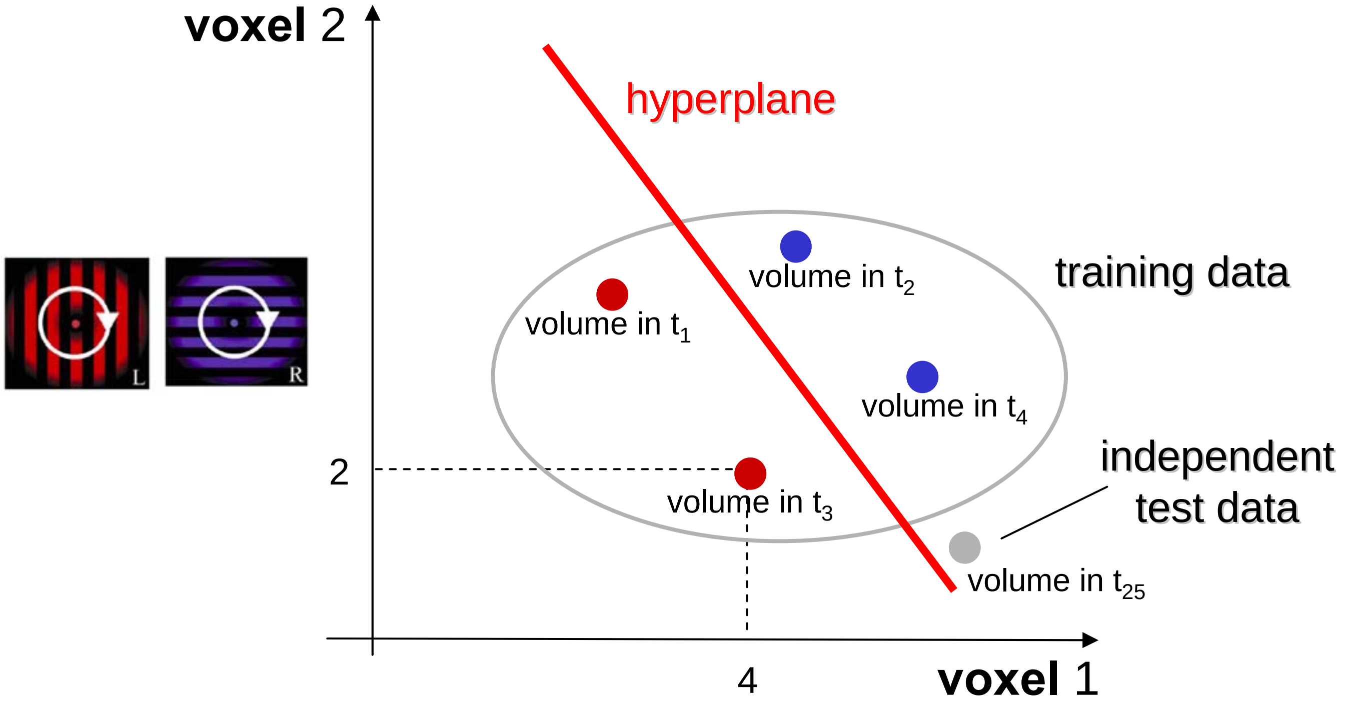
- create new features as a function of existing features
- Linear functions (PCA, ICA,…)
- Nonlinear functions during classification (i.e. hidden units in a neural network)

Outline

- WHY PATTERN CLASSIFICATION?
- PROCESSING STREAM
- PREPROCESSING / FEATURE REDUCTION
- **CLASSIFICATION**
- EVALUATING RESULTS
- APPLICATIONS

Classification

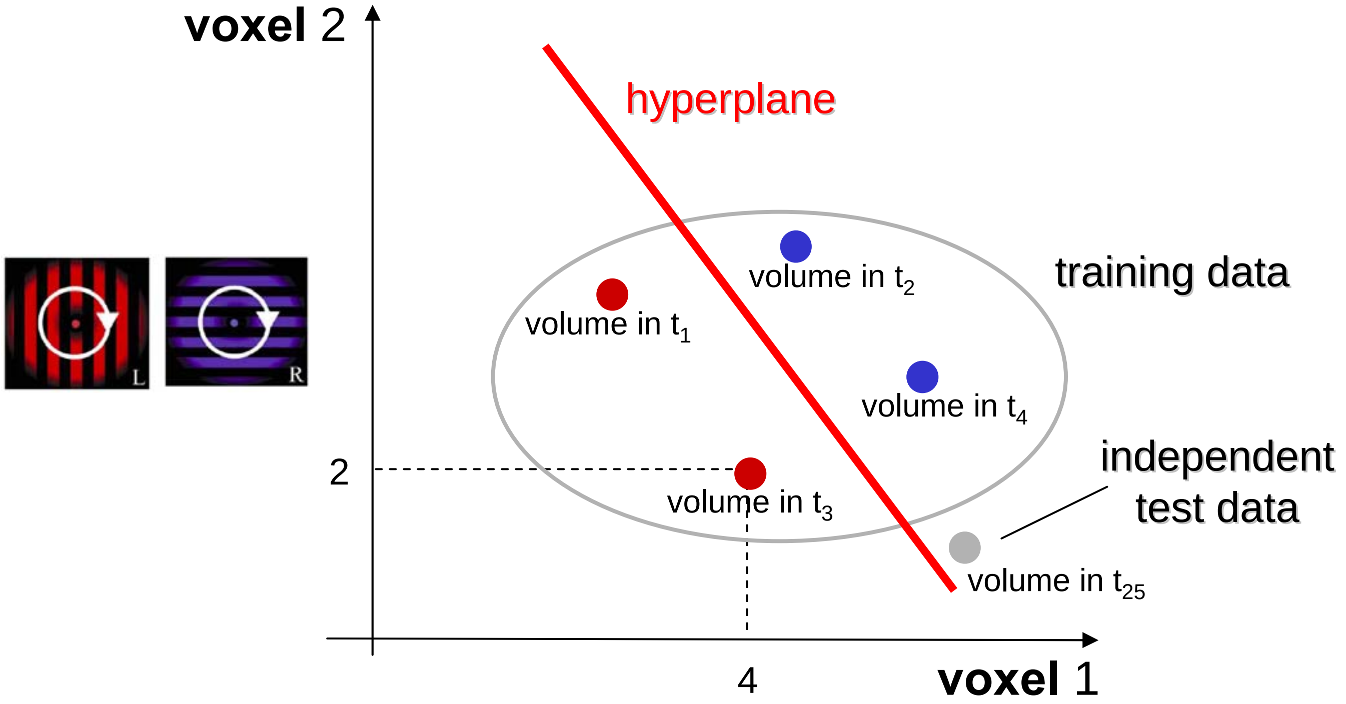
Linear classification



- ➢ our task: find a *hyperplane* that separates both conditions

Classification

Linear classification

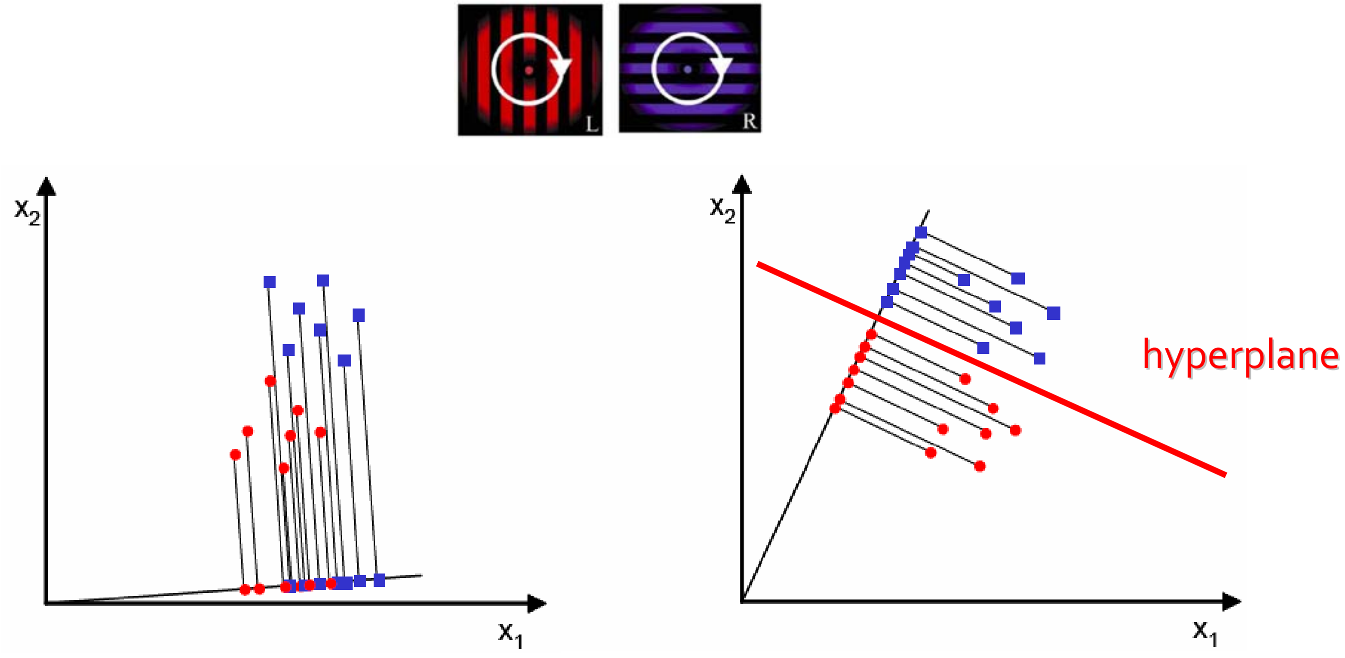


decision function: $y = f(x) = w_1 x_1 + w_2 x_2 + \ldots + w_n x_n + b$

- if y < 0, predict red // if y > 0, predict blue
- prediction = linear function of features

Classification

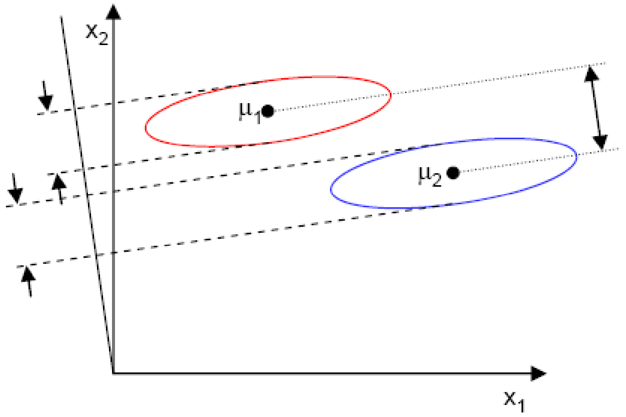
Linear classification



- Project data on a new axis that maximes the class separability
- Hyperplane is orthogonal to the best projection axis

Classification

Simplest Approach: Fisher Linear Discriminant (FLD)



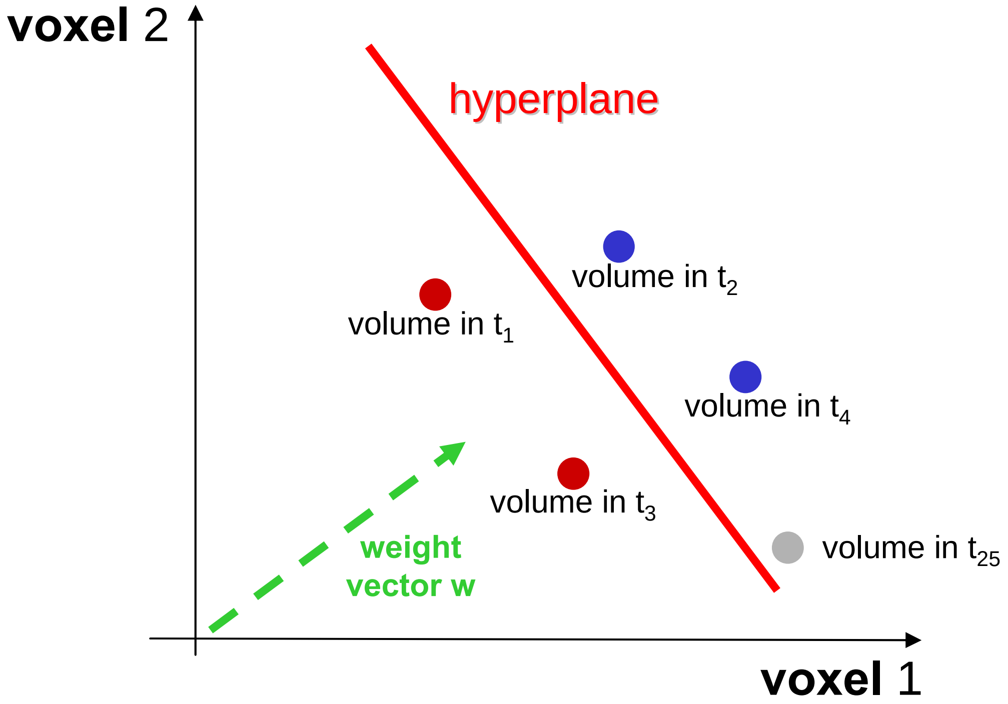
- FLD classifies by projecting the training set on the axis that is defined by the difference between the center of mass for both classes, corrected by the within class scatter

- separation is maximised for: $w = \dfrac{m_1 - m_2}{\mathrm{cov}_{class1} + \mathrm{cov}_{class2}}$

Classification

SPM Course Edinburgh 2010

Linear classification

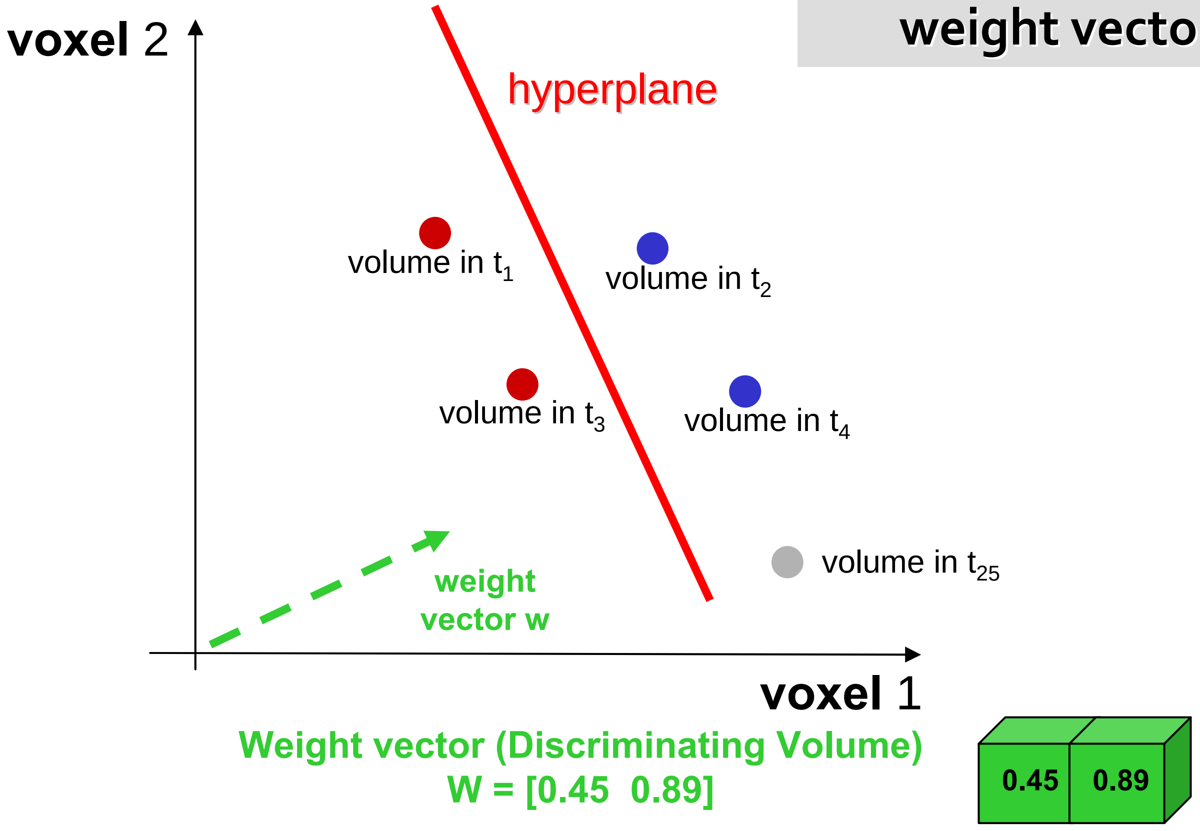


$y = \mathbf{wx} + b$ → hyperplane defined by weight vector w and offset b

Classification

Linear classification

How to interpret the weight vector?

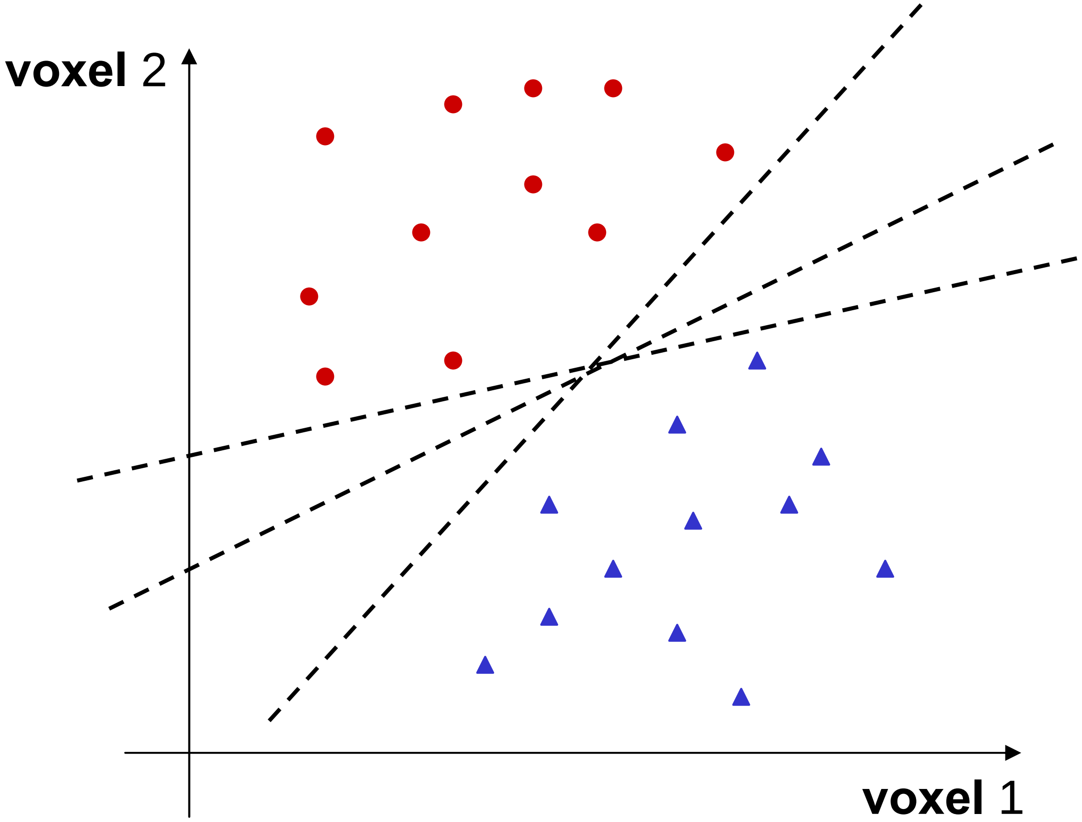


Weight vector (Discriminating Volume)
W = [0.45 0.89]

- The value of each voxel in the weight vector indicates its importance in discriminating between the two classes (i.e. cognitive states).

Classification

Support Vector Machine (SVM)

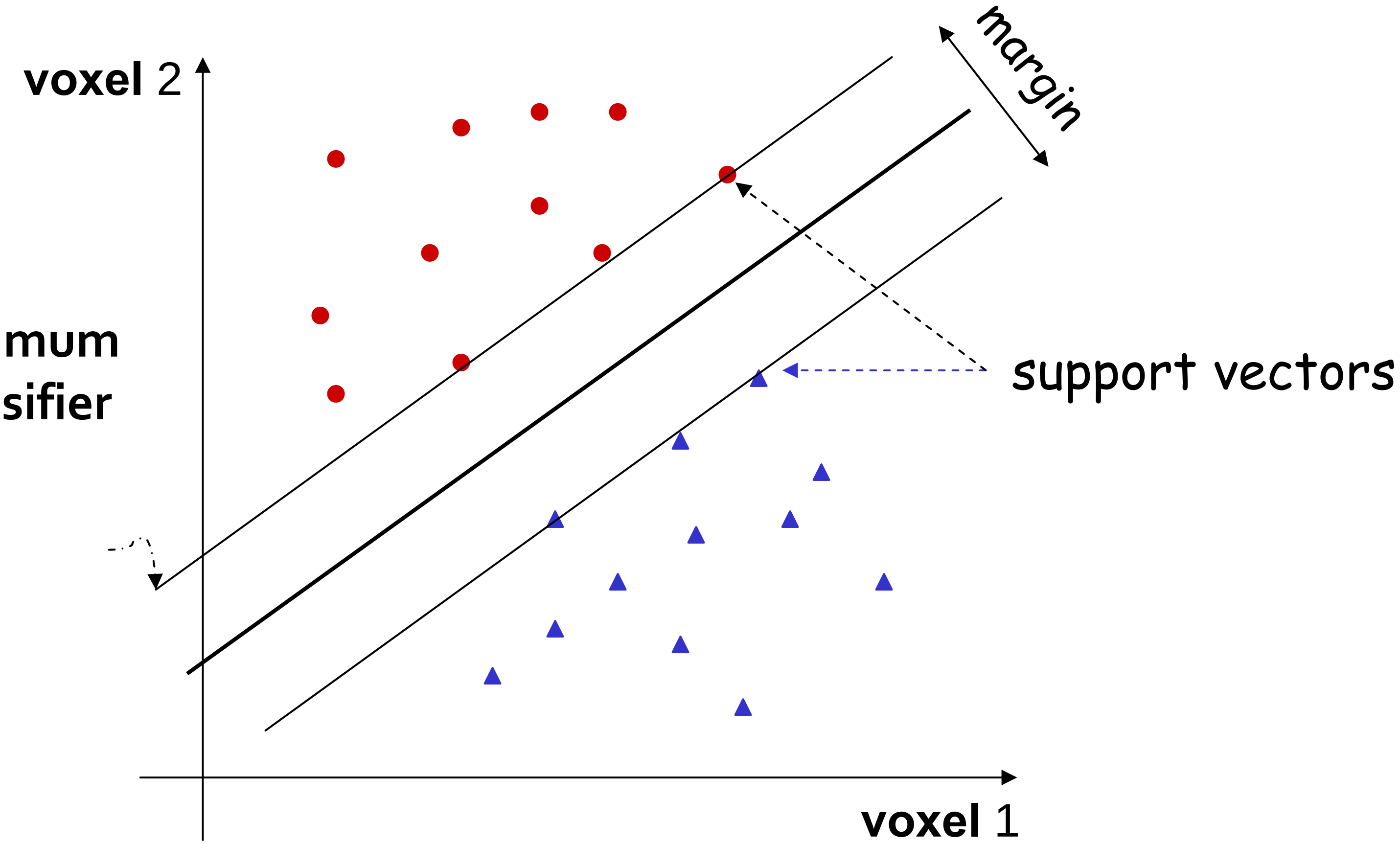


Which of the linear separators is the optimal one?

Classification

Support Vector Machine (SVM)

SVM = maximum margin classifier



If classes have overlapping distributions), SVM's are modified to account for misclassification errors by introducing additional slack variables

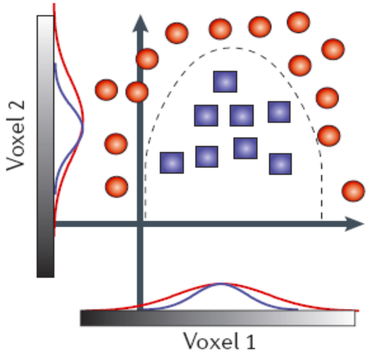
Classification

Linear classifiers

- Fisher Linear Discriminant
- Support Vector Machine (SVM)
- Logistic Regression
- Gaussian Naive Bayes
- …

Nonlinear classifiers

- SVM with kernel
- Neural Networks
- …

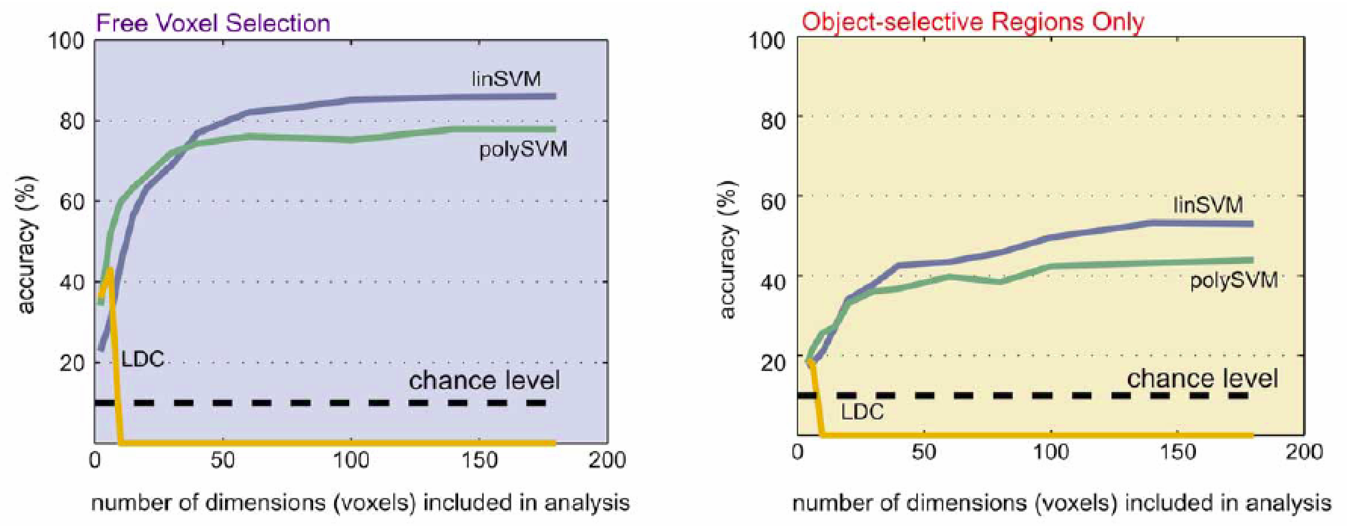


How to choose the right classifier?

Classification

Situation 1: scans ↓, features ↑ (i.e. whole brain data)

- FLD unsuitable: depends on reliable estimation of covariance matrix
- GNB inferior to SVM and LR ➔ the latter come with regularisation that help weigh down the effects of noisy and highly correlated features



Cox & Savoy (2003). NeuroImage

Classification

Situation 2: scans ↓, features ↓ (i.e. feature selection or feature extraction)

- GNB, SVM and LR: often similar performance
- SVM originally designed for two-class problems only
- SVM for multiclass problems: multiple binary comparisons, voting scheme to identify classes

- accuracy of SVM increases faster than GNB when the number of scans increase
- see Mitchell et al. (2005) for further comparisons between different classifiers

Classification

Peeking #2

- classifier performance = unbiased estimate of classification accuracy

➔ how well would the classifier label a new example randomly drawn from the same distribution?

- testing a trained classifier needs to be performed on a dataset the classifier has never seen before

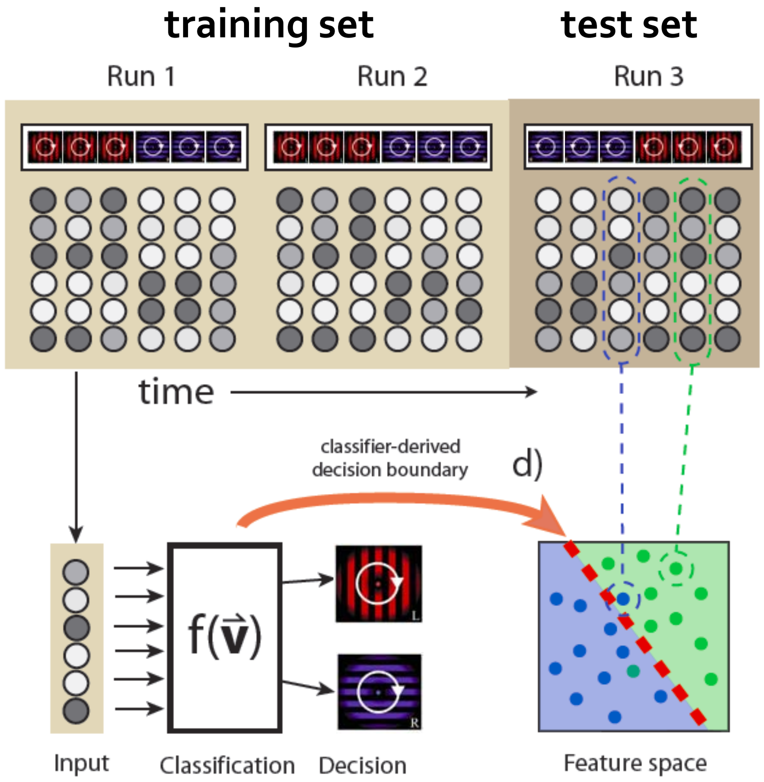
➔ if entire dataset is used for training a classifier, classification estimates become overly optimistic

Solution: leave-one-out crossvalidation

Classification

Crossvalidation

- standard approach: leave-one-out crossvalidation
- split dataset into n folds (i.e. runs)
- train classifier on 1:n-1 folds
- test the trained classifier on fold n
- rerun training/testing while withholding a different fold
- repeat procedure until each fold has been withheld once
- Classification accuracy usually computed as mean accuracy



Outline

- WHY PATTERN CLASSIFICATION?
- PROCESSING STREAM
- PREPROCESSING / FEATURE REDUCTION
- CLASSIFICATION
- **EVALUATING RESULTS**
- APPLICATIONS

Evaluating results

Can I publish my data with 57% classification accuracy in Science or Nature?

Independent test data

- Classification accuracy = unbiased estimate of the true accuracy of the classifier
- Question: what is the probability of obtaining 57% accuracy under the null hypothesis (no information about the variable of interest in my data)?
- Binary classification: p-value can be calculated under a binomial distribution with N trials (i.e. 100) and P probability of success (i.e. 0.5)
- Matlab: p = 1 - binocdf(X,N,P) = 0.067 (hmm...)

  X = number of correctly labeled examples (i.e. 57)

Evaluating results

Nonparametric approaches

Permutation tests (i.e. Polyn et al, 2005):

- create a null distribution of performance values by repeatedly generating scrambled versions of the classifier output
- MVPA: wavelet based scrambling technique (Bullmore et al., 2004)
  ➔ can accomodate non-independent data

Bootstrapping

- estimate the variance and distribution of a statistic (i.e. voxel weights)
- Multiple iterations of data resampling by drawing with replacement from the dataset

Multiclass problems: accuracy can be painful

- ➢ average rank of the correct label
- ➢ average of all pairwise comparisons

Getting results

<u>Design considerations</u>

- acquire as many <u>training</u> examples as possible ➔ classifier needs to be able to „see through the noise"
- averaging consecutive TR's can help to reduce the impact of noise (but may also eliminate natural, informative variation)
- alternative to averaging: use beta weights from a GLM analysis (i.e. based on FIR or HRF) ➔ requires many runs / trials
- avoid using consecutive scans for training a classifier ➔ lots of highly similar datapoints do not give new information
- acquire as many <u>test</u> examples as possible ➔ increases the power of significance test
- balance conditions ➔ if not, classifier may tend to focus on predominant condition

Outline

- ➢ WHY PATTERN CLASSIFICATION?
- ➢ PROCESSING STREAM
- ➢ PREPROCESSING / FEATURE REDUCTION
- ➢ CLASSIFICATION
- ➢ EVALUATING RESULTS
- ➢ APPLICATIONS

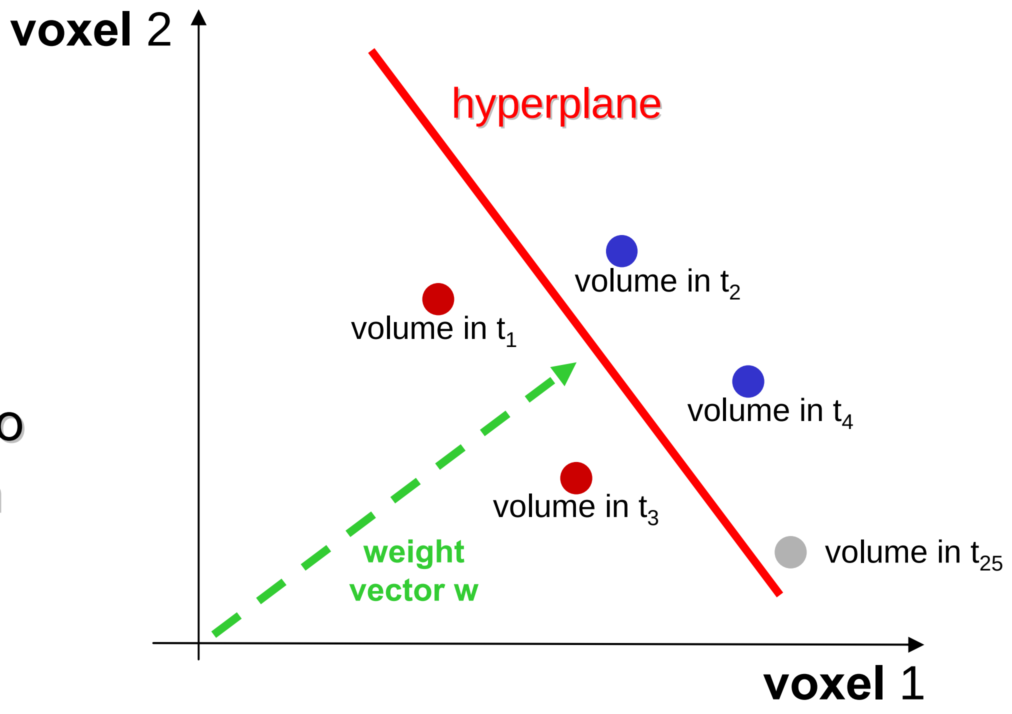
Applications

Pattern discrimination

- Question 1: do the selected fMRI data contain information about a variable of interest (i.e. conscious percept in Haynes & Rees)?

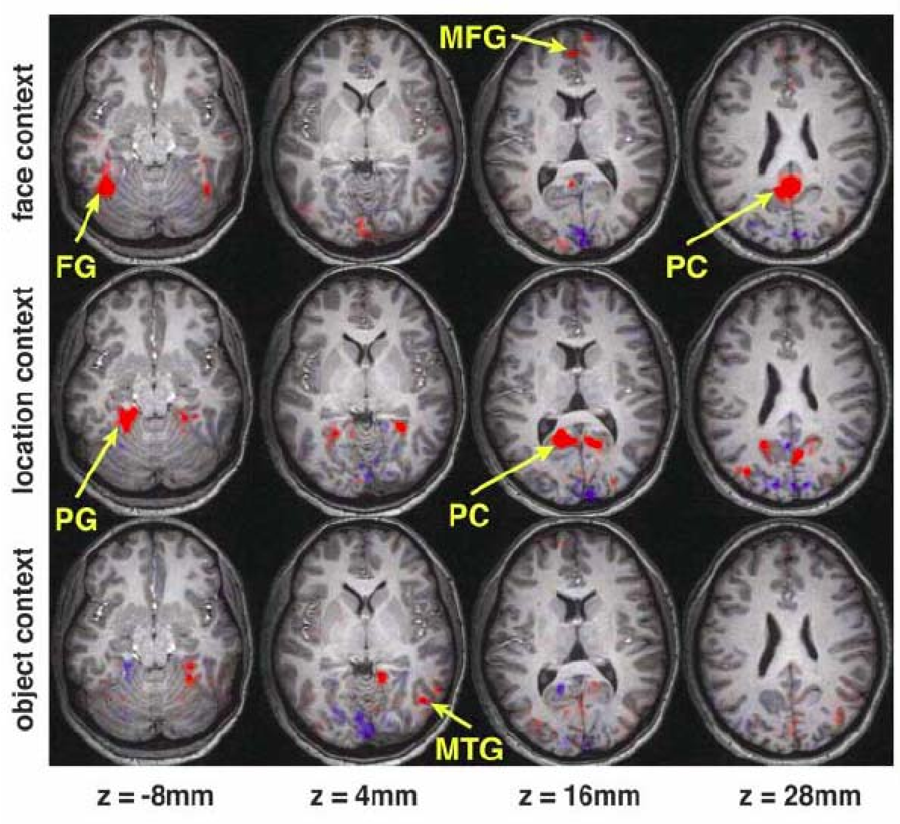
Pattern localization

- Question 2: where in the brain is information about the variable of interest represented?
- weight vector contains info on the importance of each voxel for differentiating between classes



Applications

Pattern localization - Space

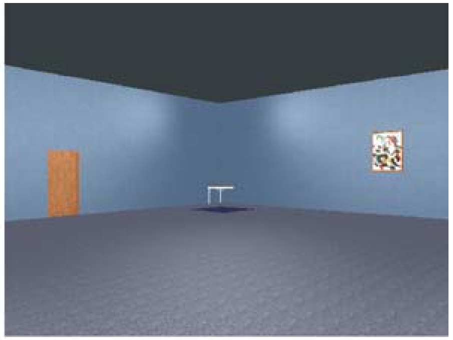


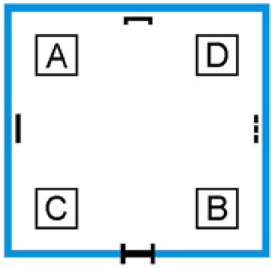
Polyn et al. (2005), Science.

Applications

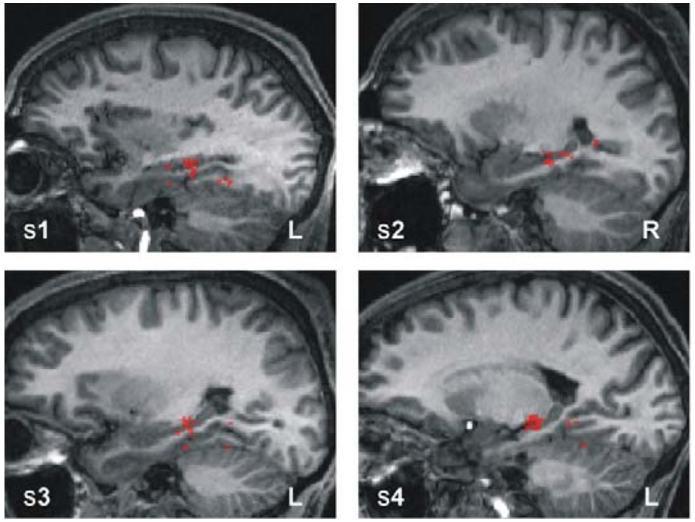
Pattern localization - Space

- *Searchlight analysis*: classification/crossvalidation is performed on a voxel and its (spherical) neighbourhood
- classification accuracy is assigned to centre voxel
- searchlight is moved across entire dataset to obtain accuracy estimates for each voxel
- can be used for feature selection or to generate a brain map of p-values





position → class.



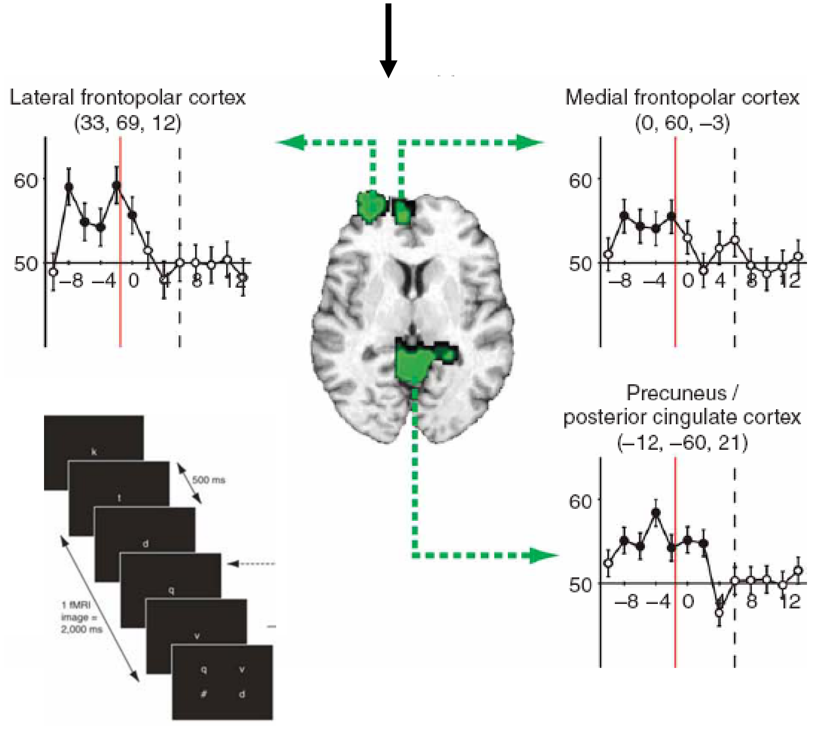
Hassabis et al. (2009), Current Biology.

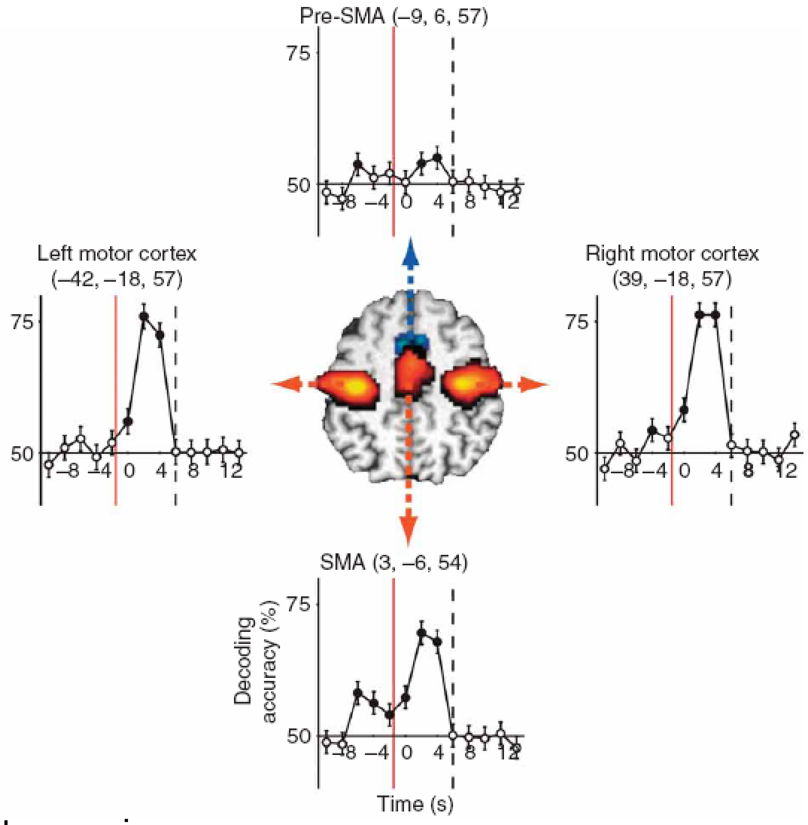
Applications

Pattern localization - Time

Question 3: when does the brain represent information about different classes?

Motor intention



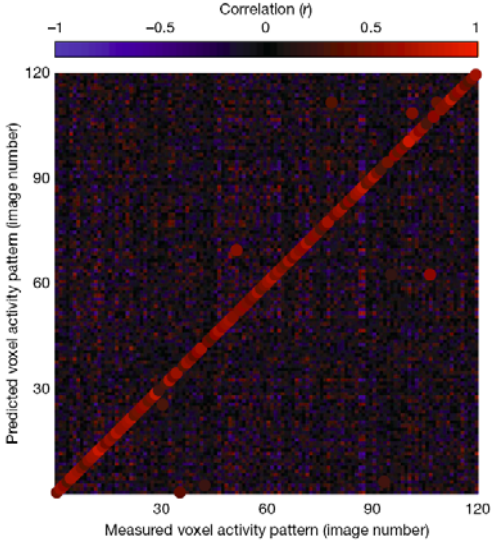


Soon et al. (2008), Nature Neuroscience.

Applications

Pattern characterization

- Question 4: How are stimulus classes represented in the brain?
- goal: characterizing the relationship between stimulus classes and BOLD patterns
- Kay et al. (2008): training of a receptive field model for each voxel in V1, V2 and V3 based on location, spatial frequency and orientation (1750 natural images)



- subsequent classification of completely new stimuli (120 natural images)

Topics

Useful literature

- Haynes JD, Rees G (2006) Decoding mental states from brain activity in humans. *Nat Rev Neurosci* 7:523-534.
- Formisano E, De Martino F, Valente G (2008) Multivariate analysis of fMRI time series: classification and regression of brain responses using machine learning. Magn Reson Imaging 26(7):921-34.
- Kriegeskorte N, Goebel R, Bandettini P (2006) Information-based functional brain mapping. *Proc Natl Acad Sci* U S A 103:3863-3868.
- Mitchell TM, et al. (2004) Learning to Decode Cognitive States from Brain Images. *Machine Learning* 57:145-175.
- Norman KA, Polyn SM, Detre GJ, Haxby JV (2006) Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends Cogn Sci* 10:424-430.
- O'Toole et al. (2007). Theoretical, statistical, and practical perspectives on pattern-based classification approaches to the analysis of functional neuroimaging data. J Cogn Neurosci.19(11):1735-52
- Pereira F, Mitchell TM, Botvinick M (2009) Machine Learning Classifiers and fMRI: a tutorial overview. *Neuroimage* 45(1 Suppl):S199-209.