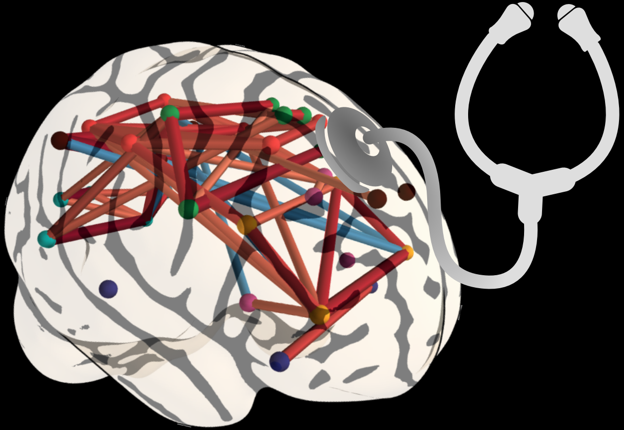
Machine learning for functional connectomes

Gaël Varoquaux Inria PARIETAL

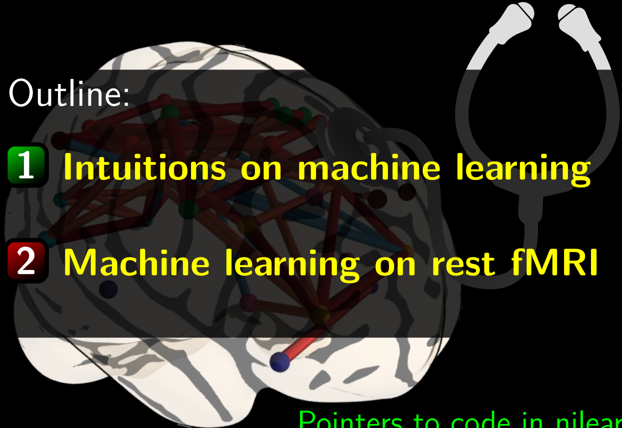


Machine learning for functional connectomes

Gaël Varoquaux

Inria

PARIETAL



Outline:

1 **Intuitions on machine learning**

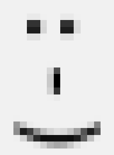
2 **Machine learning on rest fMRI**

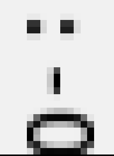
Pointers to code in nilearn & scikit-learn
nilearn.github.io — scikit-learn.org
Use the “API reference” to look up functions
and scroll down for examples of usage

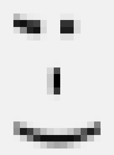
1 Intuitions on machine learning

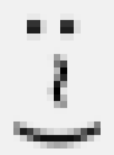
Adjusting models for prediction

Face recognition


Andrew

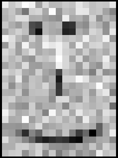

Bill

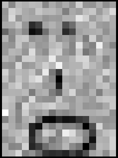

Charles

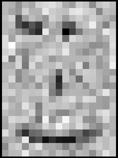

Dave

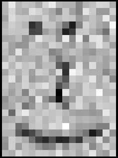
1 Machine learning in a nutshell: an example

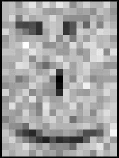
Face recognition


Andrew


Bill


Charles


Dave


?

1 Machine learning in a nutshell

A simple method:

1 Store all the known (noisy) images and the names that go with them.

2 From a new (noisy) images, find the image that is most similar.

"Nearest neighbor" method

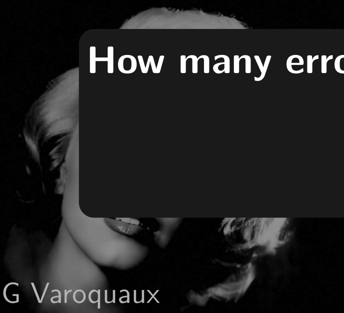
1 Machine learning in a nutshell

A simple method:

1 Store all the known (noisy) images and the names that go with them.

2 From a new (noisy) images, find the image that is most similar.

"Nearest neighbor" method



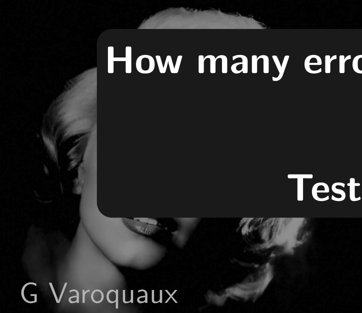
How many errors on already-known images?

1 Machine learning in a nutshell

A simple method:

1. Store all the known (noisy) images and the names that go with them.
2. From a new (noisy) images, find the image that is most similar.

"Nearest neighbor" method

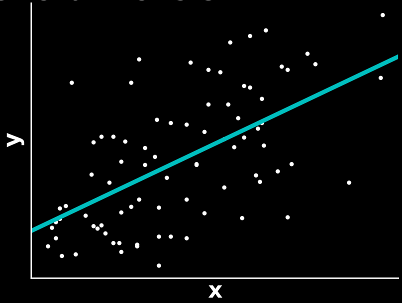


How many errors on already-known images?

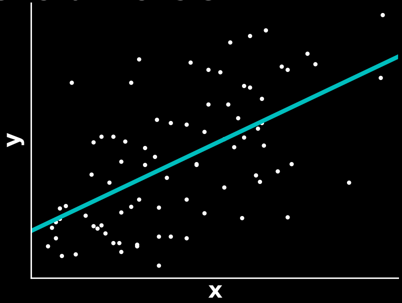
... 0: no errors

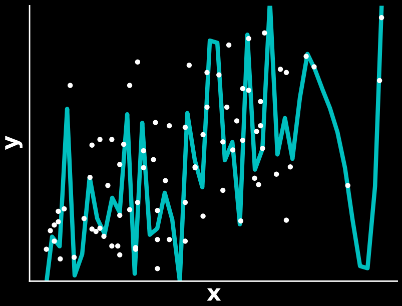
Test data $\neq$ Train data

A single descriptor: one dimension



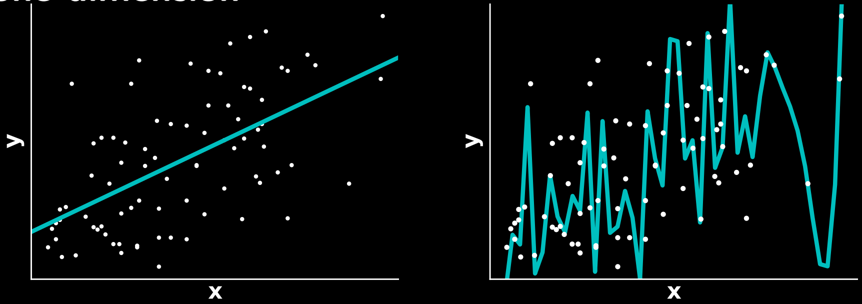
A single descriptor: one dimension





Which model to prefer?

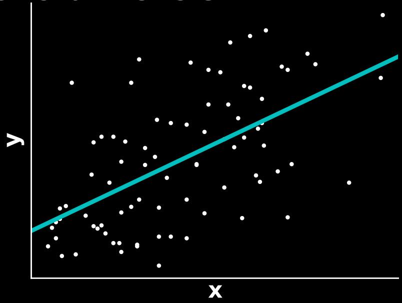
A single descriptor: one dimension

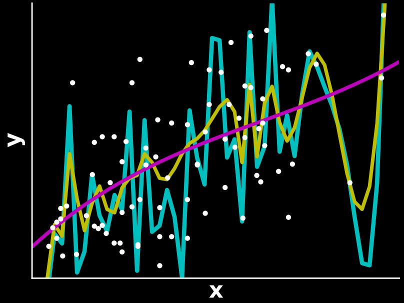


Problem of *"over-fitting"*

- Minimizing error is not always the best strategy (learning noise)
- Test data $\neq$ train data

A single descriptor: one dimension



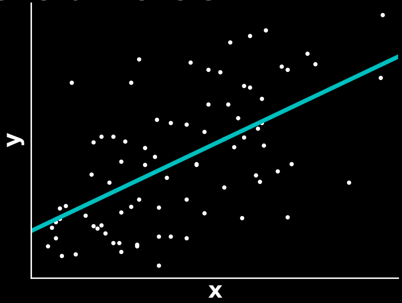


Prefer simple models

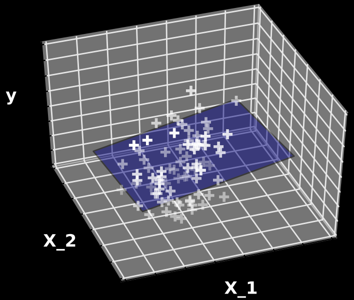
= concept of *"regularization"*

Balance the number of parameters to learn
with the amount of data

A single descriptor: one dimension



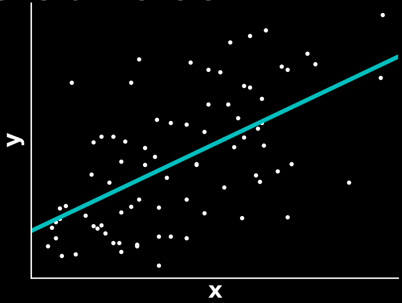
Two descriptors: 2 dimensions



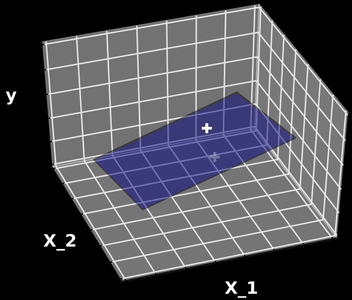
The higher the number of descriptors

the more the trouble

A single descriptor: one dimension



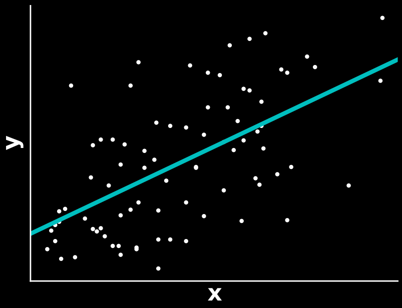
Two descriptors: 2 dimensions

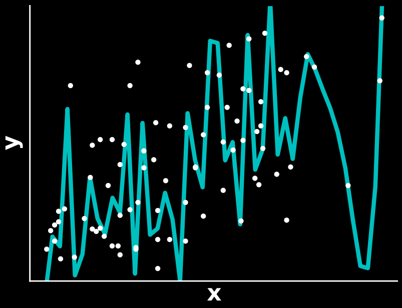


The higher the number of descriptors

the more the trouble

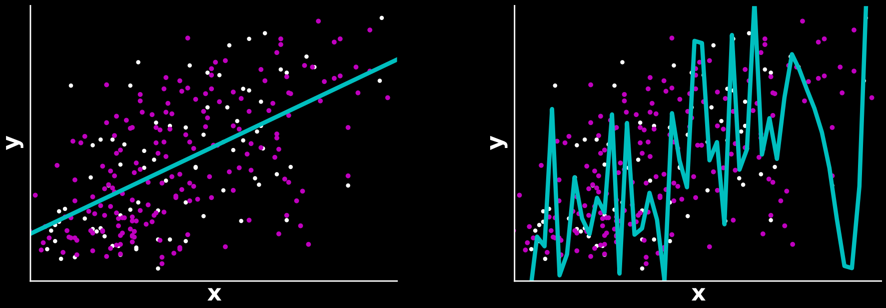
The higher the required number of subjects



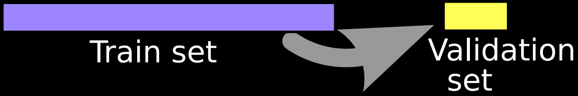


[Varoquaux... 2017]

1 **Testing prediction**: generalization and cross-validation



⇒ Need test on **independent, unseen data**

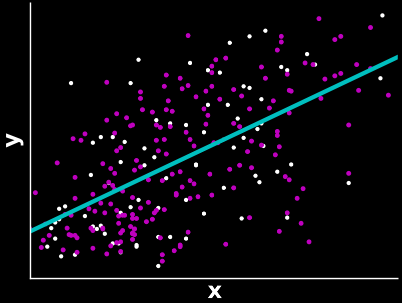


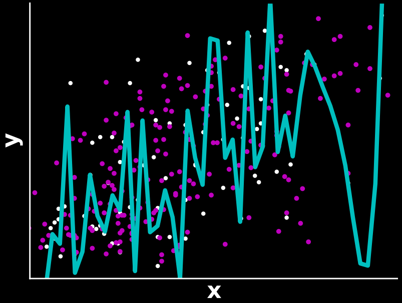
Measures prediction accuracy

sklearn.model_selection.train_test_split

[Varoquaux... 2017]

1 **Testing prediction**: generalization and cross-validation

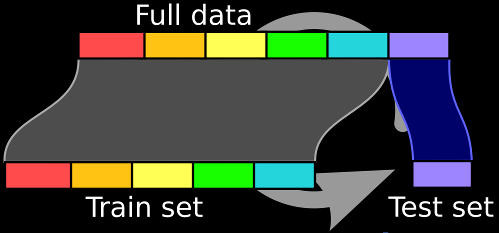




⇒ Need test on **independent, unseen data**

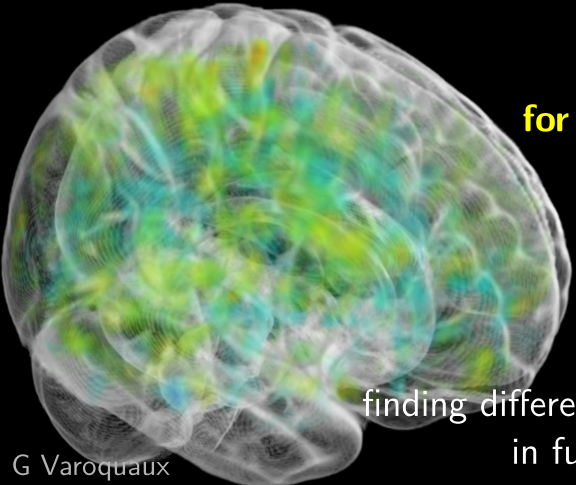
Loop

sklearn.
model_selection.
cross_val_score



[Varoquaux... 2017]

2 Machine learning on rest fMRI

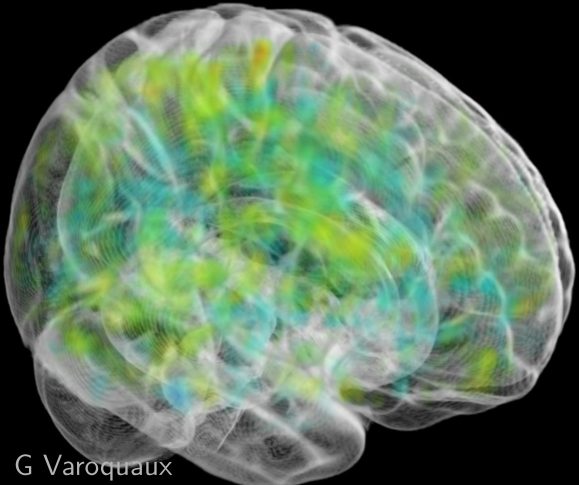


for population imaging

finding differences between subjects in functional connectomes

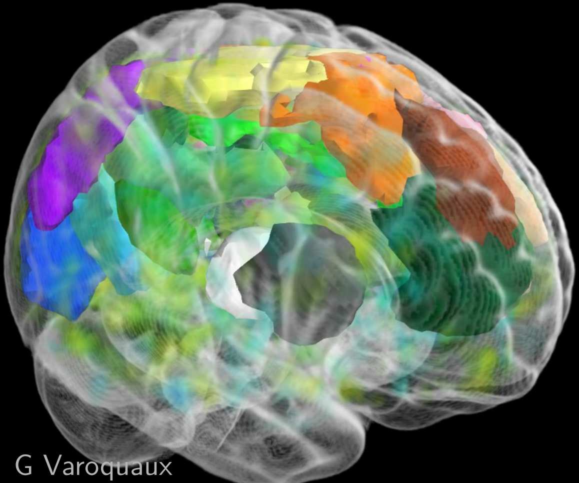
From rest-fMRI to biomarkers

No salient features in rest fMRI



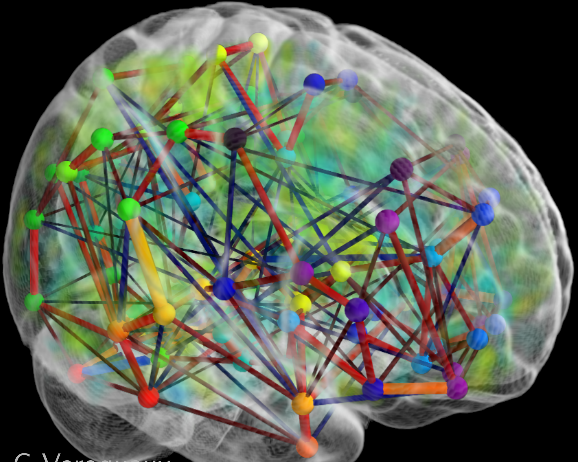
From rest-fMRI to biomarkers

- Define functional regions

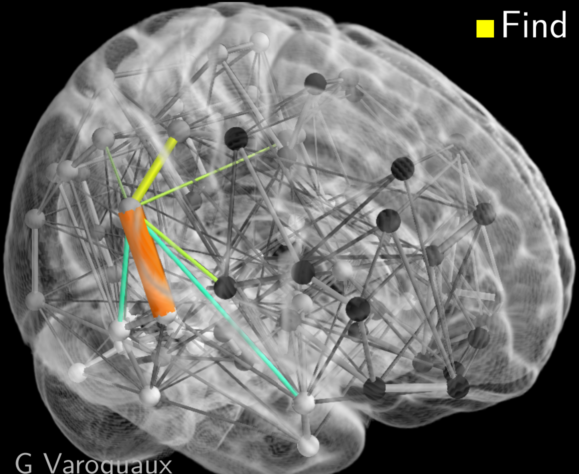


From rest-fMRI to biomarkers

- Define functional regions
- Learn interactions

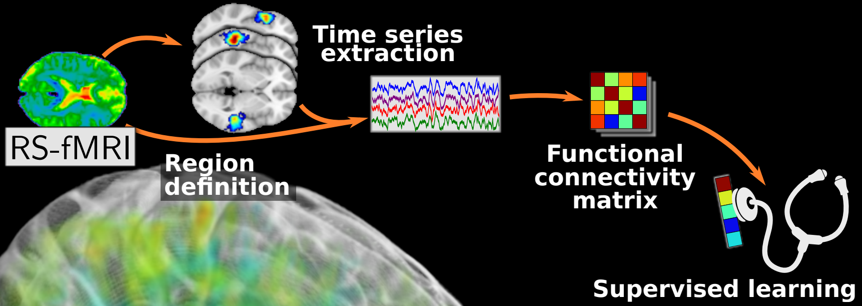


From rest-fMRI to biomarkers



- Define functional regions
- Learn interactions
- Find differences

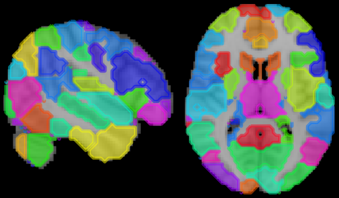
From rest-fMRI to biomarkers



Typical pipeline [Varoquaux and Craddock 2013]

1. Define regions
2. Extract times series
3. Build functional-connectivity matrix
4. Apply supervised machine learning

Clustering nilearn.regions.Parcellations



k-means

- Fast (in nilearn)
- No spatial model

$\Rightarrow$ smooth the data

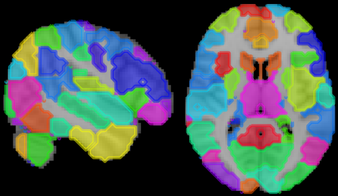
2 Defining regions from rest-fMRI

Clustering nilearn.regions.Parcellations

k-means

- Fast (in nilearn)
- No spatial model

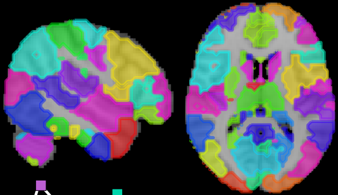
$\Rightarrow$ smooth the data

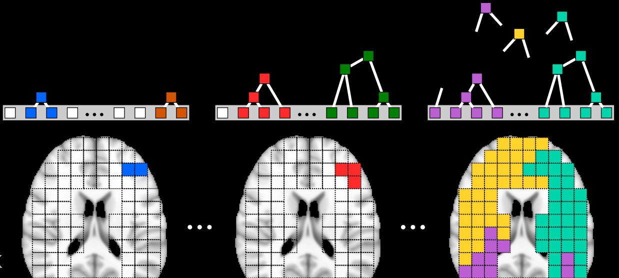


Ward agglomerative clustering

- Recursive merges of clusters
- Spatial model constraints merges

$\Rightarrow$ fast





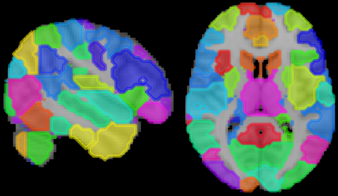
2 Defining regions from rest-fMRI

Clustering nilearn.regions.Parcellations

k-means

- Fast (in nilearn)
- No spatial model

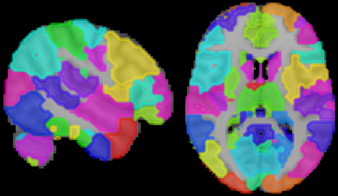
$\Rightarrow$ smooth the data



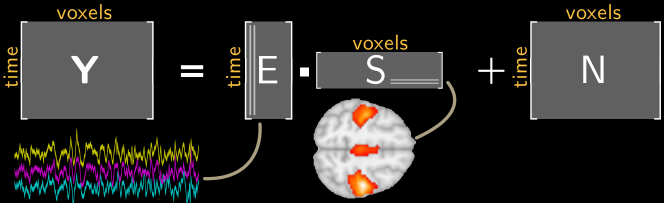
Ward agglomerative clustering

- Recursive merges of clusters
- Spatial model constraints merges

$\Rightarrow$ fast



Decomposition models

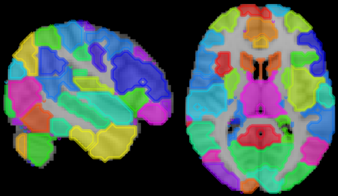


Clustering nilearn.regions.Parcellations

k-means

- Fast (in nilearn)
- No spatial model

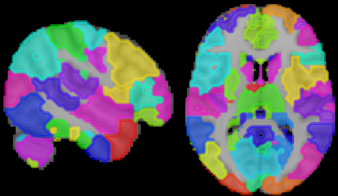
⇒ smooth the data



Ward agglomerative clustering

- Recursive merges of clusters
- Spatial model constraints merges

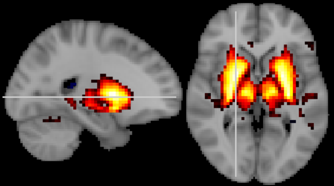
⇒ fast



Decomposition models

- ICA: nilearn.decomposition.CanICA

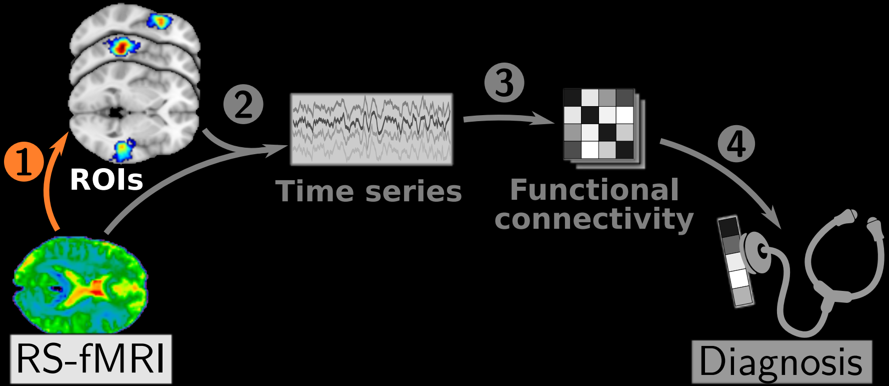
  seek independence of maps



- Sparse dictionary learning:

  seek sparse maps

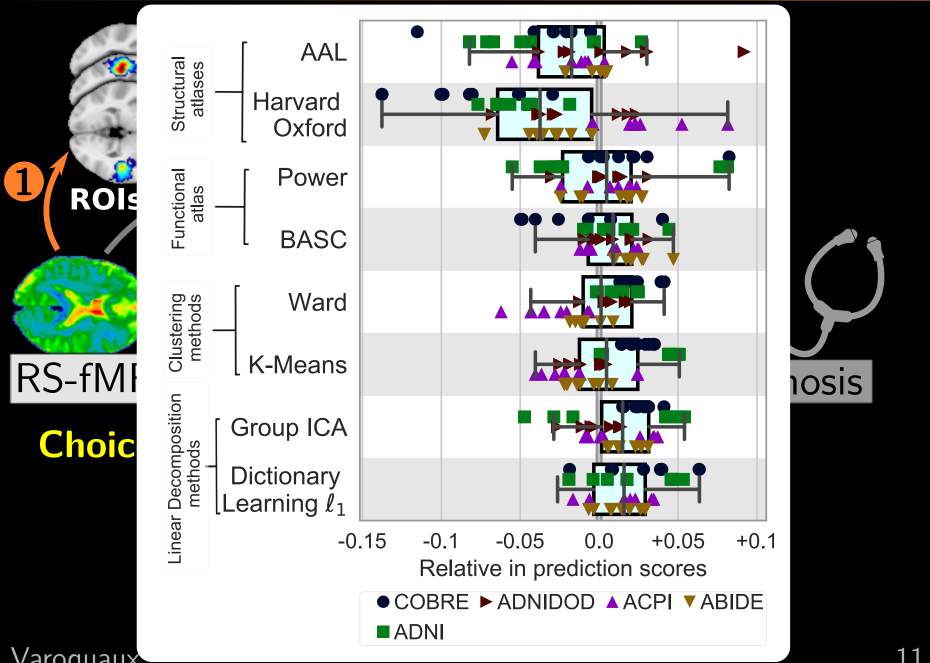
  nilearn.decomposition.DictLearning



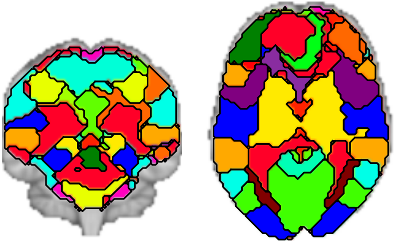
Choice of regions for best prediction?

2 For connectome prediction

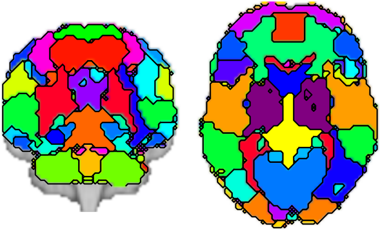
[Dadi... 2018]



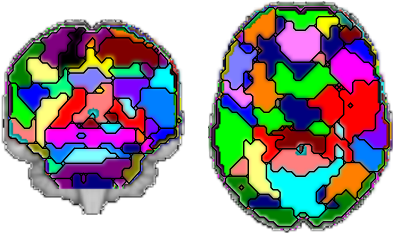
2 Region definition: resulting parcellations



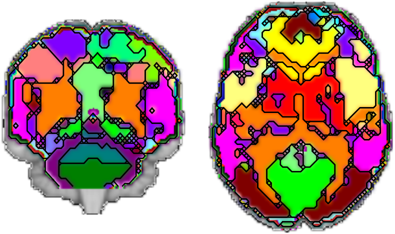
Dictionary learning



Group ICA

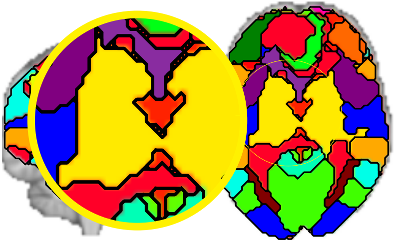


Ward clustering

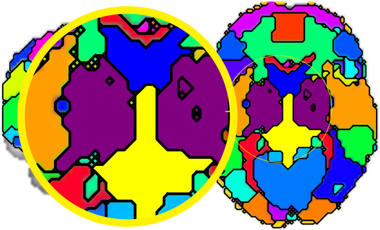


K-Means clustering

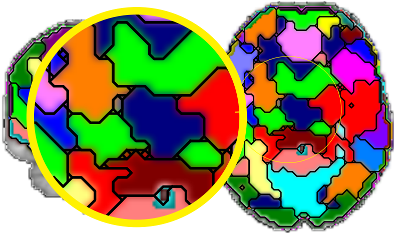
2 Region definition: resulting parcellations



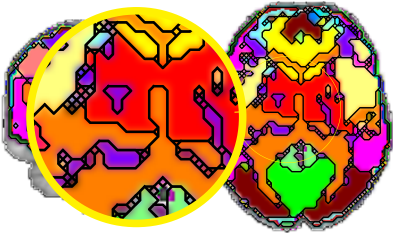
Dictionary learning



Group ICA

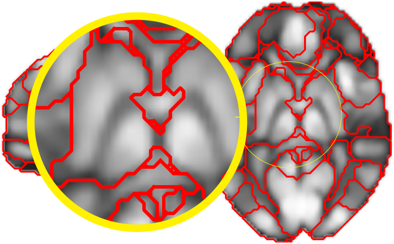


Ward clustering

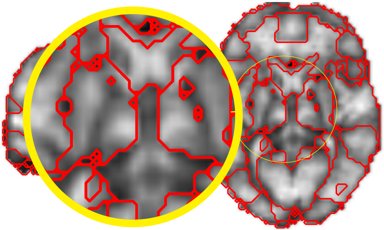


K-Means clustering

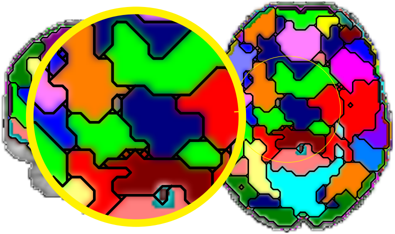
2 Region definition: resulting parcellations



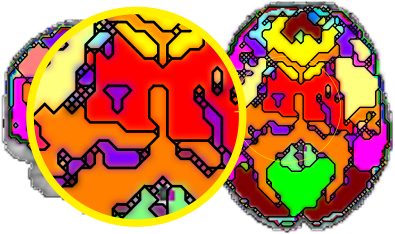
Dictionary learning



Group ICA



Ward clustering



K-Means clustering

2 Time-series extraction

- Extract ROI-average signal:

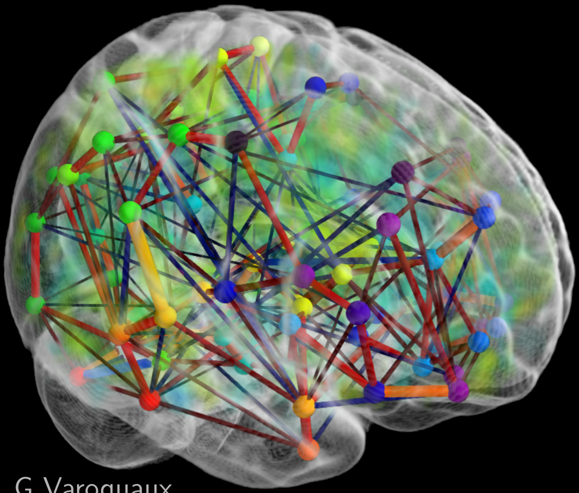
- **Optional** low-pass filter
  (≈ .1 Hz – .3 Hz)

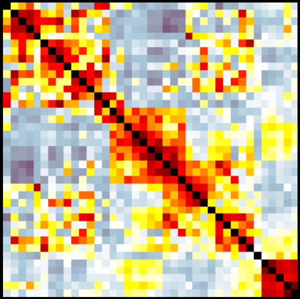
- Regress out confounds (movement parameters, CSF & white matter signals, Compcorr, Global mean)

Hard parcellations (eg from clustering)
nilearn.input_data.NiftiLabelsMasker

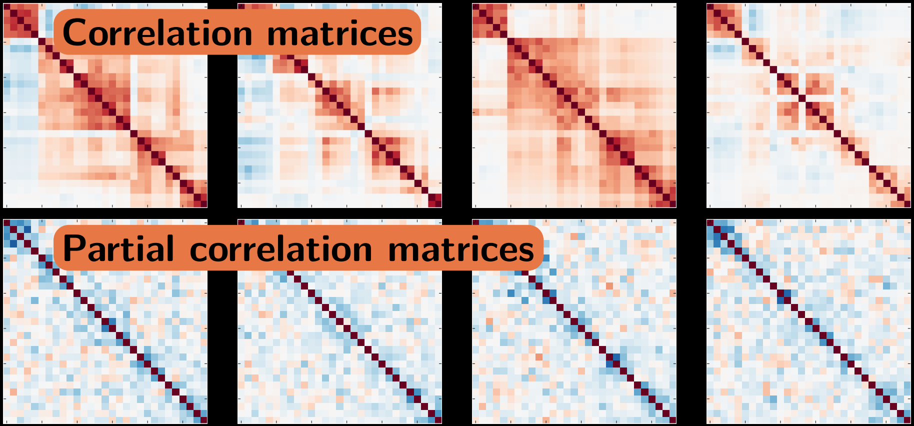
Soft parcellations (eg from ICA)
nilearn.input_data.NiftiMapsMasker

How to capture and represent interactions?





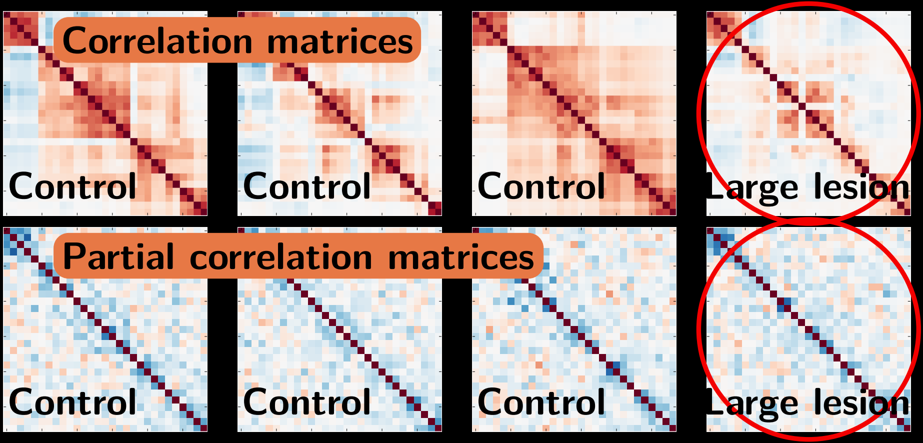
2 Connectome: differences across subjects



3 controls, 1 severe stroke patient

Which is which?

2 Connectome: differences across subjects



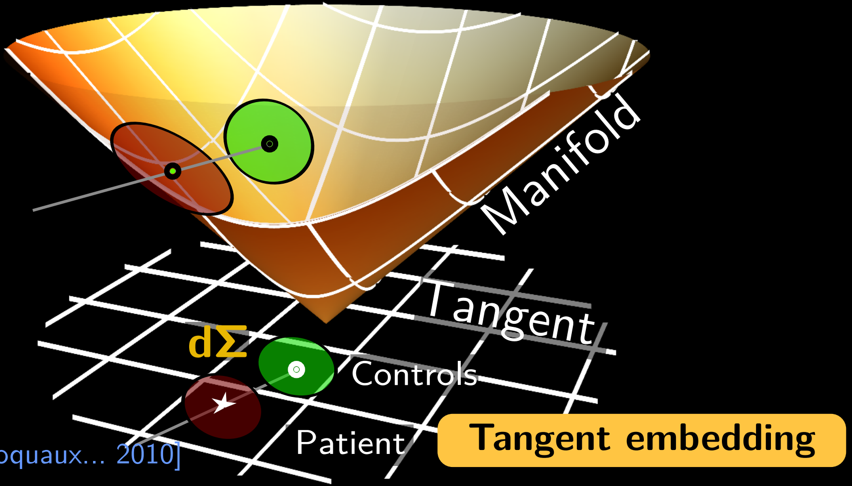
- Spread-out variability in correlation matrices
- Noise in partial-correlations

Strong dependence between coefficients

[Varoquaux... 2010]

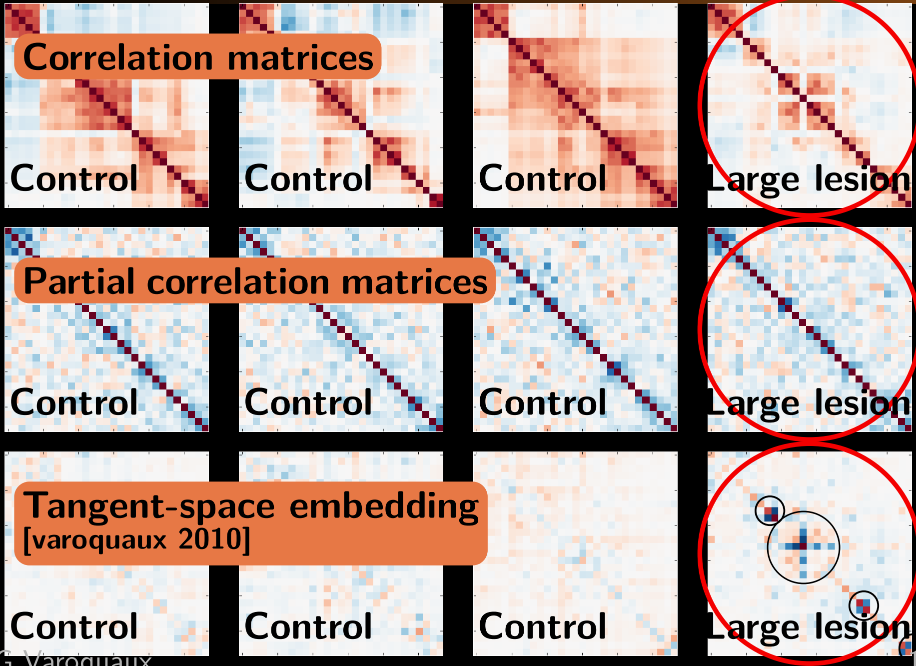
2 **Information geometry**: uniform-error parametrization

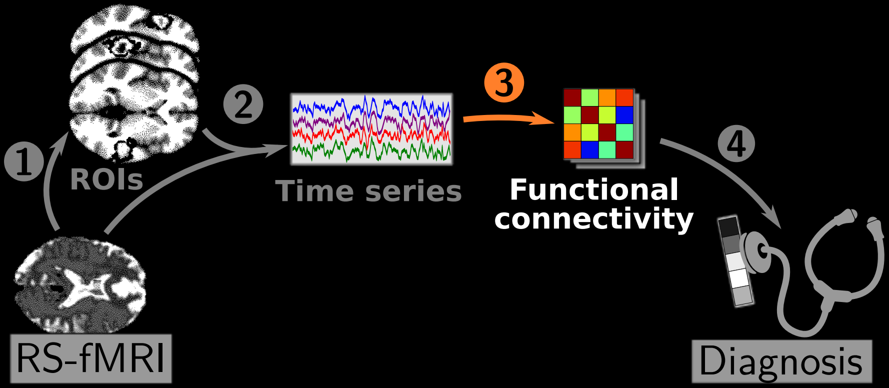
- Subject-specific noise in covariance form manifold
- Tangent space removes coupling in coefficients



[Varoquaux... 2010]

2 **Connectome**: which parametrization maps differences?

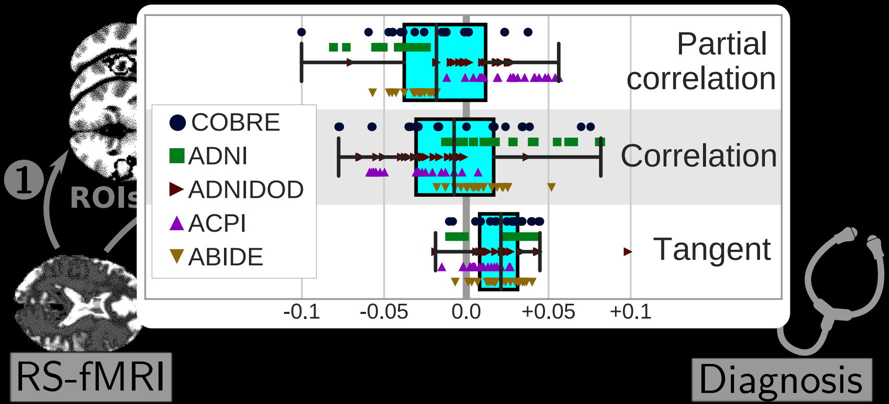




Connectivity matrix

nilearn.connectome.ConnectivityMeasure

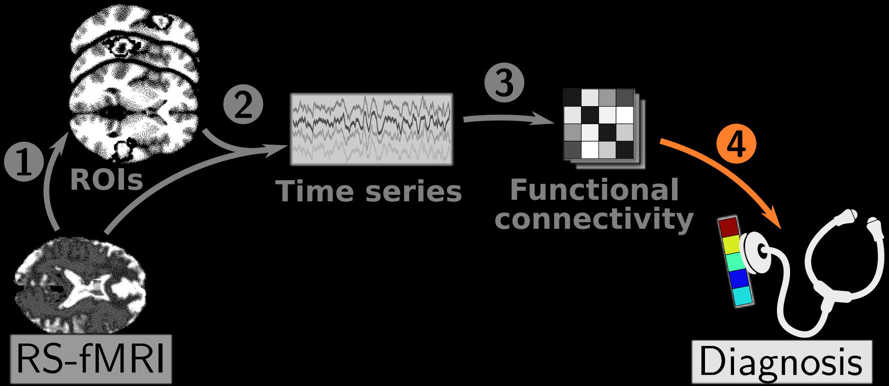
- Correlation
- Partial correlations
- Tangent space



Connectivity matrix

- Correlation
- Partial correlations
- Tangent space

nilearn.connectome.ConnectivityMeasure

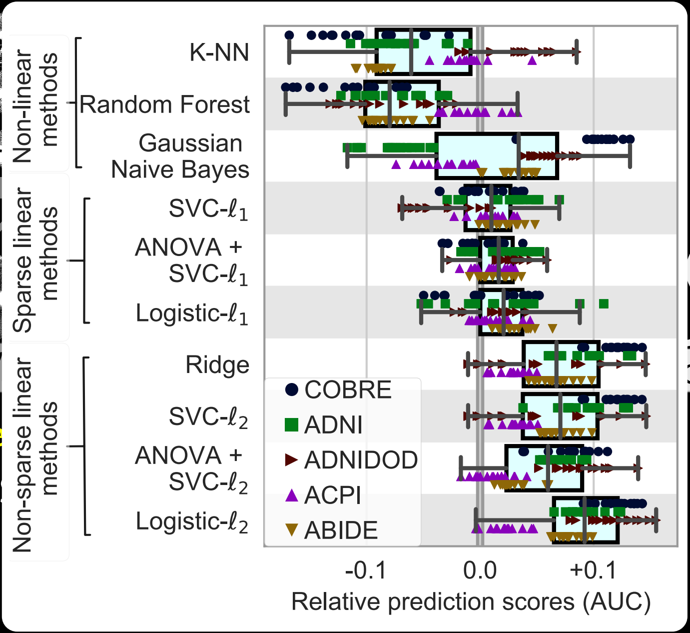


Supervised learning

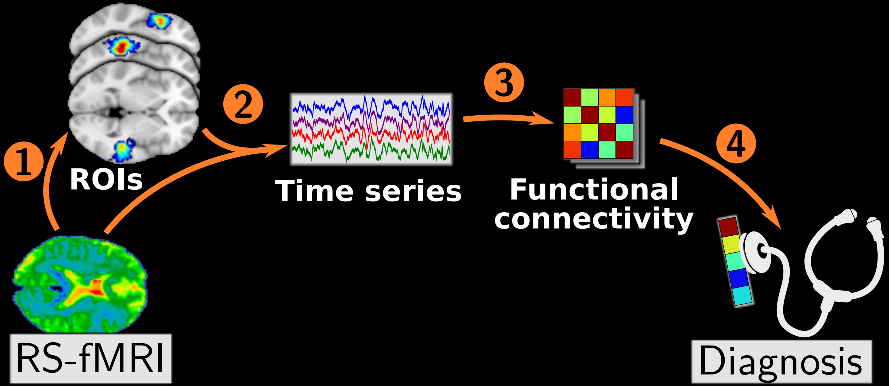
- Stick with Linear models
  sklearn.linear_model.LogisticRegression

2 Supervised learning step

[Dadi... 2018]



Predicting from brain activity at rest



1. Functional regions (eg clustering, decomposition, or BASC atlas)
2. Filtering and or confound removal
3. Tangent-space parametrization
4. Supervised linear models (*eg* SVMs)

A. Abraham, E. Dohmatob, B. Thirion, D. Samaras, and G. Varoquaux. Extracting brain regions from rest fMRI with total-variation constrained dictionary learning. In *MICCAI*, page 607. 2013.

K. Dadi, M. Rahim, A. Abraham, D. Chyzhyk, M. Milham, B. Thirion, and G. Varoquaux. Benchmarking functional connectome-based predictive models for resting-state fmri. 2018.

G. Varoquaux and R. C. Craddock. Learning and comparing functional connectomes across subjects. *NeuroImage*, 80:405, 2013.

G. Varoquaux and B. Thirion. How machine learning is shaping cognitive neuroimaging. *GigaScience*, 3:28, 2014.

G. Varoquaux, F. Baronnet, A. Kleinschmidt, P. Fillard, and B. Thirion. Detection of brain functional-connectivity difference in post-stroke patients using group-level covariance modeling. In *MICCAI*. 2010.

3 References II

G. Varoquaux, P. R. Raamana, D. A. Engemann, A. Hoyos-Idrobo, Y. Schwartz, and B. Thirion. Assessing and tuning brain decoders: cross-validation, caveats, and guidelines. *NeuroImage*, 145:166–179, 2017.