An introduction to machine learning for fMRI

Francisco Pereira

Botvinick Lab
Princeton Neuroscience Institute
Princeton University
what is machine learning?

- how to build computer systems that automatically improve with experience
what is machine learning?

- how to build computer systems that automatically improve with experience

- building models of data for
  - predicting numeric variables (regression)
  - predicting categoric variables (classification)
  - grouping data points (clustering)
  - …
what is machine learning?

- how to build computer systems that automatically improve with experience

- building models of data for
  - predicting numeric variables (regression)
  - predicting categoric variables (classification)
  - grouping data points (clustering)
  - ...

- overlaps with applied statistics
why use it at all?

to tell a story about data

[adapted from slides by Russ Poldrack]
Once upon a time there was a sample...
... and then came a beautiful model...

fit model by estimating parameters

RT = 503.7 - age**-0.530

[adapted from slides by Russ Poldrack]
very suggestive, but...

Is RT really related to age?

RT = 503.7 + age$^*{-0.530}$

[adapted from slides by Russ Poldrack]
is RT related to age?

Model:
\[ RT = \beta_0 + \beta_1 \text{age} + \varepsilon \]

parameters in the population

RT = 503.7 + age*−0.530

[adapted from slides by Russ Poldrack]
is RT related to age?

Model:
\[ RT = \beta_0 + \beta_1 \text{age} + \varepsilon \]

parameters in the population

parameters estimated from the sample

\[ \beta_{\text{est}} = (X'X)^{-1}X'y \]

[adapted from slides by Russ Poldrack]
is RT related to age?

Model:
\[ RT = \beta_0 + \beta_1 \times \text{age} + \varepsilon \]

- **parameters** in the population
- **parameters** estimated from the sample
  \[ \beta_{\text{est}} = (X'X)^{-1}X'y \]

Hypothesis:
Is \( \beta_1 \) different from 0?

[adapted from slides by Russ Poldrack]
is RT related to age?

Null hypothesis: $\beta_1 = 0$
Alternative: $\beta_1 \neq 0$

$RT = 503.7 + \text{age} \times -0.530$
is RT related to age?

Null hypothesis: $\beta_1 = 0$
Alternative: $\beta_1 \sim= 0$

How likely is the parameter estimate ($\beta_1 = -0.53$) if the null hypothesis is true?

$RT = 503.7 + age^{*-0.530}$
is RT related to age?

Null hypothesis: $\beta_1 = 0$
Alternative: $\beta_1 \neq 0$

How likely is the parameter estimate ($\beta_1 = -0.53$) if the null hypothesis is true?

We need a statistic with a known distribution to determine this!

[adapted from slides by Russ Poldrack]
is RT related to age?

the CLT tells us

$$\hat{\beta}_1 \sim N(\beta_1, Var(\hat{\beta}_1))$$

but we don’t know

$$Var(\hat{\beta}_1)$$

RT = 503.7 + age^{+0.530}
is RT related to age?

\[
\text{RT} = 503.7 + \text{age}^{*} - 0.530
\]

the CLT tells us

\[
\hat{\beta}_1 \sim N(\beta_1, \text{Var}(\hat{\beta}_1))
\]

but we don’t know

\[
\text{Var}(\hat{\beta}_1)
\]

we do know

\[
t = \frac{\hat{\beta}_1}{\sqrt{\text{Var}(\hat{\beta}_1)}} \sim T_{N-p}
\]

[adapted from slides by Russ Poldrack]
is RT related to age?

\[ t(10) = -4.76 \]

How likely is this value in this \( t \) distribution?

\[ p < .001 \]

[adapted from slides by Russ Poldrack]
what can we conclude?

- in **this** sample
  - $p < 0.001$ - there is a relationship between age and RT
  - $R^2$ - age accounts for 69% of variance in RT

- very unlikely if no relationship in the population

- the test **does not** tell us how well we can predict RT from age in the population

[adapted from slides by Russ Poldrack]
what happens with a new sample?

draw a new sample from the same population

compute the R2 using parameters estimated in the original sample
what happens with a new sample?

repeat this 100 times…

using model parameters estimated from the original sample

average $R^2 = 0.578$

a measure of how good the model learned from a single sample is

[adapted from slides by Russ Poldrack]
the learning perspective

When we estimate parameters from a sample, we are learning about the population from training data.
the learning perspective

When we estimate parameters from a sample, we are learning about the population from training data.

How well can we measure the prediction ability of our learned model? Use a new sample as test data.
test data and cross-validation

If you can’t collect more split your sample in two...

[adapted from slides by Russ Poldrack]
test data and cross-validation

If you can’t collect more split your sample in two...

k-fold cross-validation:
- split into k folds
- train on k-1, test on the left out
- average prediction measure on all k folds
- several variants: all possible splits, leave-one-out

[adapted from slides by Russ Poldrack]
leave-one-out cross-validation

regression lines on each training set

original sample
$R^2 = 0.694$

leave-one-out on original
$R^2 = 0.586$

mean over 100 new samples
$R^2 = 0.591$

[adapted from slides by Russ Poldrack]
model complexity

As model complexity goes up, we can always fit the training data better

What does this do to our predictive ability?

[adapted from slides by Russ Poldrack]
model complexity

polynomials of higher degree fit the training data better...

[adapted from slides by Russ Poldrack]
polynomials of higher degree fit the training data better...

... but they do worse on test data: overfitting

[adapted from slides by Russ Poldrack]
model complexity

if the relationship in the population were more complicated

[adapted from slides by Russ Poldrack]
if the relationship in the population were more complicated we could use CV to determine adequate model complexity (this would need to be done with nested CV, CV inside the training set)
what is machine learning, redux

- generalization: make predictions about a new individual

- a model that generalizes captures the relationship between the individual and what we want to predict

- cross-validation is a good way of
  - measuring generalization
  - doing model selection (there are others)
what is machine learning, redux

- generalization: make predictions about a new individual

- a model that generalizes captures the relationship between the individual and what we want to predict

- cross-validation is a good way of
  - measuring generalization
  - doing model selection (there are others)

- “all models are wrong but some are useful”
  George Box
what does this have to do with fMRI?

In this talk:

- prediction is classification
- generalization is within subject (population of trials)
- how to draw conclusions with statistical significance
- what has it been used for?
two questions

GLM: are there voxels that reflect the stimulus?

stimulus → fMRI activation (single voxel)

contrast of interest → GLM
two questions

**GLM:** are there voxels that reflect the stimulus?

stimulus → fMRI activation (single voxel)

contrast of interest → GLM

**Classifier:** do voxels contain information to predict?

stimulus ← fMRI activation (multiple voxels)

what is the subject seeing?
case study: two categories

[data from Rob Mason and Marcel Just, CCBI, CMU]

- subjects read concrete nouns in 2 categories
  - words name either tools or building types
  - trial:
    - see a word
    - think about properties, use, visualize
    - blank

3 seconds
8 seconds
case study: two categories

- subjects read concrete nouns in 2 categories
  - words name either tools or building types
  - trial:
    - see a word
    - think about properties, use, visualize
    - blank
    
    3 seconds
    
    8 seconds

- goal: can the two categories be distinguished?
- average images around trial peak to get one labelled image
the name(s) of the game

average trial image → example

voxels (features) → class label

tools
the name(s) of the game

average trial image

example

voxels (features) class label

tools

14 examples

example group

tools labels =

...
the name(s) of the game

average trial image

example

voxels (features) class label

tools

14 examples

dataset

example group

tools labels =

84 examples

dataset

group 1

…

group 6
classifying two categories

training data (42)   labels

<table>
<thead>
<tr>
<th>group 1</th>
<th></th>
<th>...</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>group 3</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 5</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 2</th>
<th></th>
<th>...</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>group 4</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 6</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
classifying two categories

training data (42)       labels

- group 1
- group 3
- group 5

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

test data (42)

- group 2
- group 4
- group 6

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

+ classifier
classifying two categories

training data (42)  

<table>
<thead>
<tr>
<th>group 1</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

labels

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>

predicted labels  vs  true labels

test data (42)  

<table>
<thead>
<tr>
<th>group 2</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
classifying two categories

<table>
<thead>
<tr>
<th>training data (42)</th>
<th>labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>group 1</td>
<td></td>
</tr>
<tr>
<td>group 3</td>
<td></td>
</tr>
<tr>
<td>group 5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>test data (42)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>group 2</td>
<td></td>
</tr>
<tr>
<td>group 4</td>
<td></td>
</tr>
<tr>
<td>group 6</td>
<td></td>
</tr>
</tbody>
</table>

predicted labels vs true labels

accuracy estimate

= \#correct/42

= 0.65
a classifier

- is a function from **data** to **labels**
- parameters learned from **training data**
a classifier

- is a function from **data** to **labels**
- parameters learned from **training** data

- want to estimate its **true accuracy**
  - “probability of labelling a new example correctly”
- estimation is done on the **test** data
  - it is finite, hence an estimate with uncertainty
a classifier

- is a function from **data** to **labels**
- parameters learned from **training data**

- want to estimate its **true accuracy**
  - “probability of labelling a new example correctly”
- estimation is done on the **test** data
  - it is finite, hence an estimate with uncertainty

- null hypothesis: “the classifier learned nothing”
what questions can be tackled?

- is there information?
  (pattern discrimination)
- where/when is information present?
  (pattern localization)
- how is information encoded?
  (pattern characterization)
“is there information?”

- what is inside the black box?
- how to test results?
- from a study to examples
what is inside the box?

- simplest function is no function at all
- “nearest neighbour”
what is inside the box?

- simplest function is no function at all
- “nearest neighbour”
what is inside the box?

- simplest function is no function at all
- “nearest neighbour”
what is inside the box?

- simplest function is no function at all
- “nearest neighbour”

Requires example similarity measure.

Euclidean, correlation, …
what is inside the box?

- next simplest: learn linear discriminant
what is inside the box?

- next simplest: learn linear discriminant
- note that there are many solutions...
what is inside the box?

- next simplest: learn linear discriminant
- note that there are many solutions...
what is inside the box?

- next simplest: learn linear discriminant
- note that there are many solutions...
linear classifiers

\[
\text{If } \sum weight_i x_i > 0 \text{ tools }
\]

otherwise \text{ buildings}
linear classifiers

If \( \sum_{i=0}^{n} \text{weight}_i \cdot x_i > 0 \) tools

otherwise buildings

various kinds

Gaussian Naive Bayes

Logistic Regression

Linear SVM

differ on how weights are chosen
linear classifiers

If \( \sum \text{weight}_i \times \text{voxel}_i \) > 0, tools

otherwise, buildings

linear SVM weights:
linear classifiers

If

\[ \text{weight}_0 + \text{weight}_1 x + \text{weight}_2 x + \ldots + \text{weight}_n x > 0 \]

Tools

otherwise

Buildings

linear SVM weights:

weights pull towards tools

weights pull towards buildings
nonlinear classifiers

- linear on a transformed feature space
nonlinear classifiers

- linear on a transformed feature space

- neural networks:
  new features are learnt
nonlinear classifiers

- linear on a transformed feature space!

- neural networks:
  new features are learnt

- SVMs
  new features are (implicitly) determined by a kernel

64
nonlinear classifiers

reasons to be careful:

- too few examples,
  too many features
- harder to interpret
nonlinear classifiers

reasons to be careful:

- too few examples,
  too many features
- harder to interpret

- overfitting

![Graph showing prediction error vs. increasing model complexity with best tradeoff between test set and training set.](from Hastie et al, 2001)
how do we test predictions?

training data (42)  labels

+ classifier = predicted labels vs true labels

test data (42)

<table>
<thead>
<tr>
<th>group 1</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 3</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 5</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 2</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 4</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>group 6</th>
<th></th>
<th>...</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
how do we test predictions?

Predicted labels

tools
buildings
buildings

...

tools
buildings
tools
### how do we test predictions?

<table>
<thead>
<tr>
<th>True labels</th>
<th>Predicted labels</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>tools</td>
<td>tools</td>
<td></td>
</tr>
<tr>
<td>tools</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>buildings</td>
<td>tools</td>
<td></td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>tools</td>
<td>tools</td>
<td></td>
</tr>
</tbody>
</table>

#correct out of #test
how do we test predictions?

<table>
<thead>
<tr>
<th>True labels</th>
<th>Predicted labels</th>
<th>#correct out of #test</th>
</tr>
</thead>
<tbody>
<tr>
<td>tools</td>
<td>tools</td>
<td>error</td>
</tr>
<tr>
<td>tools</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>buildings</td>
<td>tools</td>
<td>error</td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
<td></td>
</tr>
<tr>
<td>tools</td>
<td>tools</td>
<td></td>
</tr>
</tbody>
</table>

- null hypothesis:
  "classifier learnt nothing" $\rightarrow$ "predicts randomly"
How do we test predictions?

<table>
<thead>
<tr>
<th>True labels</th>
<th>Predicted labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>tools</td>
<td>tools</td>
</tr>
<tr>
<td>tools</td>
<td>buildings</td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>buildings</td>
<td>tools</td>
</tr>
<tr>
<td>buildings</td>
<td>buildings</td>
</tr>
<tr>
<td>tools</td>
<td>tools</td>
</tr>
</tbody>
</table>

- null hypothesis:
  “classifier learnt nothing” → “predicts randomly”

- intuition:
  - a result is significant if very unlikely under null
how do we test predictions?

- $X = \#\text{correct}$
- $P(X|\text{null})$ is binomial($\#\text{test}, 0.5$)
- $p$-value is $P(X\geq \text{result to test}|\text{null})$

distribution under null
(0.05 $p$-value cut-off)
how do we test predictions?

- $X = \#\text{correct}
- P(X|\text{null})$ is binomial($\#\text{test}, 0.5$)
- p-value is $P(X = \text{result to test}|\text{null})$
- lots of caveats:
  - accuracy is an estimate
  - few examples $\Rightarrow$ very uncertain
  - can get a confidence interval
  - must correct for multiple comparisons

distribution under null
(0.05 p-value cut-off)
- leave-one-example-out is expensive
  - leave-one-run-out is often reasonable
  - bonus: it helps separate training from test examples
- keep a balanced #examples in the training set
summary: is there information?

- classifiers
  - nearest-neighbour, linear, nonlinear
- usage
  - learn the classifier in the training set
  - apply it to test set and predict labels
- testing
  - typical null
  - how to test it
what questions can be tackled?

- is there information? (pattern discrimination)
- where/when is information present? (pattern localization)
- how is information encoded? (pattern characterization)
case study: orientation

[Kamitani & Tong, 2005]

subjects see gratings in one of 8 orientations
case study: orientation

subjects see gratings in one of 8 orientations

voxels in visual cortex respond similarly to different orientations

voxel responses orientations

[Kamitani & Tong, 2005]
case study: orientation

subjects see gratings in one of 8 orientations

yet, voxels can be combined to predict the orientation of the grating being seen!

voxel responses orientations

voxels in visual cortex respond similarly to different orientations

linear SVM

45° detector

[Kamitani&Tong, 2005]
features

- case study #1, features are voxels
- case study #2, features are voxels in visual cortex
features

- case study #1, features are voxels
- case study #2, features are voxels in visual cortex

- what else could they be?
  voxels at particular times in a trial,
  syntactic ambiguity study
features

- you can also synthesize features
  - Singular Value Decomposition (SVD)
  - Independent Component Analysis (ICA)

\[
\text{examples} \quad \text{dataset} \quad = \quad \text{Z} \quad \text{basis images}
\]

\[
\text{voxels} \quad \text{new features}
\]
- you can also synthesize features
  - Singular Value Decomposition (SVD)
  - Independent Component Analysis (ICA)

\[
\text{examples} \quad \begin{array}{c}
\text{dataset} \\
\text{voxels}
\end{array} = Z \
\begin{array}{c}
\text{basis images} \\
\text{voxels}
\end{array} \quad \text{new features}
\]

- reduces to \#features < \#examples
- a feature has a spatial extent: basis image
- learn on the training set, convert the test set
example construction

- an example
  - can be created from one or more brain images
  - needs to be amenable to labelling
example construction

- an example
  - can be created from one or more brain images
  - needs to be amenable to labelling

- some possibilities
  - the brain image from a single TR
  - the average image in a trial or a block
  - the image of beta coefficients from deconvolution
example construction

- an example
  - can be created from one or more brain images
  - needs to be amenable to labelling

- some possibilities
  - the brain image from a single TR
  - the average image in a trial or a block
  - the image of beta coefficients from deconvolution

- caveats
  - remember the haemodynamic response time-to-peak
  - images for two examples not separate “enough”
    - in test set, lowers the effective #examples in statistical test
    - in between train and test set, “peeking”/”circularity”
  - read [Kriegeskorte et al. 2009] (”double dipping”)
localization

- key idea #1
  
  test conclusions pertain to whatever is fed to the classifier
localization

- key idea #1
  test conclusions pertain to whatever is fed to the classifier

- key idea #2
  one can predict anything that can be labelled:
  stimuli, subject percepts, behaviour, response, ...
localization

- key idea #1
  test conclusions pertain to whatever is fed to the classifier
  
- key idea #2
  one can predict anything that can be labelled:
    stimuli, subject percepts, behaviour, response, ...

so what criteria can we use?

- location
- time
- voxel behaviour or relationship to label
  - aka “feature selection”
feature (voxel) selection

- what does it look like

<table>
<thead>
<tr>
<th>training data (42)</th>
<th>labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>group 1</td>
<td>...</td>
</tr>
<tr>
<td>group 3</td>
<td>...</td>
</tr>
<tr>
<td>group 5</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>test data (42)</th>
<th>predicted labels</th>
<th>true labels</th>
</tr>
</thead>
<tbody>
<tr>
<td>group 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>group 6</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

+ classifier

= vs
feature (voxel) selection

- what does it look like

Training data (42) → Labels

Group 1
Group 3
Group 5

Test data (42)

Group 2
Group 4
Group 6

Classifier

Predicted labels vs True labels
feature (voxel) selection

- what does it look like

training data (42)  labels

group 1

group 3

group 5

test data (42)

group 2

group 4

group 6

predicted labels  true labels

=  vs
feature (voxel) selection

- what does it look like

training data (42)  

labels

group 1

group 3

group 5

test data (42)

predicted labels  true labels

=  vs

- great for improving prediction accuracy

- but
  - voxels often come from all over the place
  - very little overlap in selected across training sets
feature (voxel) selection

- look at the training data and labels
feature (voxel) selection

- look at the training data and labels
- a few criteria:
  - difference from baseline
feature (voxel) selection

- look at the training data and labels
- a few criteria:
  - difference from baseline
  - difference between classes (e.g. ANOVA)
feature (voxel) selection

- look at the training data and labels
- a few criteria:
  - difference from baseline
  - difference between classes (e.g. ANOVA)
  - preferential response to one class
feature (voxel) selection

- look at the training data and labels
- a few criteria:
  - difference from baseline
  - difference between classes (e.g. ANOVA)
  - preferential response to one class
  - stability
  - ...

![Bar charts showing data for different classes and runs.](chart.png)
case study: category reinstatement

- items from 3 categories: faces, locations, objects
- classifier trained during study of the grouped items
- detect category reinstatement during free recall (test)

faces  locations  objects

[Polyn et al 2005]
temporal localization

voxel influence on reinstatement estimate

[Polyn et al 2005]
temporal localization

key ideas:

- trained classifiers may be used as detectors to localize the time at which information is present

- test examples can be temporally overlapping

- it is feasible to decode endogenous events
classifier dissection

- a classifier learns to relate features to labels
- we can consider not just what gets selected but also how the classifier uses it
- in linear classifiers, look at voxel weights
classifier dissection

- weights depend on classifier assumptions
- less of an issue if feature selection is used

weights are similar, but accuracy difference 15%
classifier dissection

- assumption effects on synthetic data

```
6 voxels and class label

3 informative

3 inf. unequally

2 inf. 1 correlated

6 voxels and class label

-2 0 2 4

6 voxel weights and bias term

+ 

libs/vm (L=1)

0.96/0.97 (train/test)

0.89/0.89 (train/test)

0.92/0.92 (train/test)

GNB

+ 

1 2 3 4 5 6 7

0.96/0.97 (train/test)

0.88/0.99 (train/test)

0.90/0.90 (train/test)

```

“voxels”

SVM

GNB
case study: 8 categories

subjects see photographs of objects in 8 categories
- faces, houses, cats, bottles, scissors, shoes, chairs, scrambled
- block: series of photographs of the same category
case study: 8 categories

nearest neighbour classifier
- all category pair distinctions
  - fusiform gyrus
  - rest of temporal cortex
- selects voxels by behaviour
  - responsive to single category
  - responsive to multiple

logic:
- restrict by location/behaviour
- see if there is still information

[Haxby et al., 2001]
classifier dissection

1) whole-brain logistic regression weights

faces
classifier dissection

1) whole-brain logistic regression weights

faces

houses
classifier dissection

1) whole-brain logistic regression weights

faces

houses

chairs
classifier dissection

2) feature selection

faces

z=23  z=25  z=27  z=29  z=31  z=33  z=35
classifier dissection

2) feature selection

faces

z=23   z=25   z=27   z=29   z=31   z=33   z=35

top 1000 voxels

z=23   z=25   z=27   z=29   z=31   z=33   z=35
classifier dissection

whole brain classifier
- accuracy 40% in this case
- many more features than examples => simple classifier
- messy maps (can bootstrap to threshold)
classifier dissection

whole brain classifier
- accuracy 40% in this case
- many more features than examples => simple classifier
- messy maps (can bootstrap to threshold)

voxel selection
- accuracy 80% in this case
- sparse, non-reproducible maps
- different methods pick different voxels
classifier dissection

whole brain classifier
- accuracy 40% in this case
- many more features than examples => simple classifier
- messy maps (can bootstrap to threshold)

voxel selection
- accuracy 80% in this case
- sparse, non-reproducible maps
- different methods pick different voxels

a lot of work
classifier dissection

neural network
- one-of-8 classifier
- temporal cortex

learned model
- hidden units
- activation patterns across units reflect category similarity

[Hanson et al., 2004]
classifier dissection

neural network
- one-of-8 classifier
- temporal cortex

learned model
- hidden units
- activation patterns across units reflect category similarity
- sensitivity analysis
  - add noise to voxels
  - which ones lead to classification error?

[Hanson et al., 2004]
classifier dissection conclusions

- if linear works, you can look at weights
  - but know what your classifier does (or try several)
  - read about bootstrapping (and [Strother et al. 2002])
classifier dissection conclusions

- if linear works, you can look at weights
  - but know what your classifier does (or try several)
  - read about bootstrapping (and [Strother et al. 2002])

- voxel selection
  - may be necessary in multiclass situations
  - try multiple methods and **look** at the voxels they pick
  - voxels picked may be a small subset of informative ones
  - report all #s of voxels selected or use cross-validation on training set to pick a # to use
classifier dissection conclusions

- if linear works, you can look at weights
  - but know what your classifier does (or try several)
  - read about bootstrapping (and [Strother et al. 2002])

- voxel selection
  - may be necessary in multiclass situations
  - try multiple methods and look at the voxels they pick
  - voxels picked may be a small subset of informative ones
  - report all #s of voxels selected or
    use cross-validation on training set to pick a # to use

- nonlinear classifiers
  - worth trying, but try linear + voxel selection first
  - look at [Hanson et al. 2004] and [Rasmussen et al. 2011]
    for ways of gauging voxel influence on classifier
Information-based mapping (searchlights) [Kriegeskorte 2006]

- focus the classifier on small voxel neighbourhoods
- more examples than features
- can learn voxel relationships (e.g. covariance matrix)
- can train nonlinear classifiers
information mapping

- on 8 categories, yields an accuracy map

- also local information: covariance, voxel weights
information mapping

- on 8 categories, yields an accuracy map

- also local information: covariance, voxel weights
- can be thresholded for statistical significance
“H0: chance level” deems many voxels significant

what does accuracy mean in multi-class settings?
  - confusion matrix
  - for each class, what do examples belonging to it get labelled as?

accuracy 0.76

faces
houses
cats
shoes
bottles
chairs
scissors
scrambled

8 classes
information mapping

- contrast each pair of classes directly
- threshold accuracy to a binary image

face v house
face v cats
...

scrambled v chairs

count# pairs significant

voxels
information mapping

- each voxel has a *binary profile* across pairs
- how many different ones?

diagram showing data representation for different pairs:

- face vs house
- face vs cats
- scrambled vs chairs

Count # pairs significant
information mapping

- A binary profile is a kind of confusion matrix
- Only a few hundred profiles, many similar
- Cluster them!

1. Face
2. House
3. Cat
4. Bottle
5. Scissors
6. Chairs
7. Shoes
8. Scrambled

Houses versus all else

Faces and cat versus all else
information mapping

- a map of accuracy works well in 2 class situation
- some classifiers seem consistently better
  see [Pereira & Botvinick 2010] for details
- easy to get above chance with multiway prediction
  so reporting accuracy or #significant is not enough
- consider reporting common profiles or grouping classes into distinctions to test
the plot thickens

- 1-nearest neighbour correlation, all voxels
- linear classifier, all voxels
- select voxels with behaviour/location
- incorporate locality information
  - in voxels considered
  - in classifier

...
the plot thickens

- 1-nearest neighbour correlation, all voxels
- linear classifier, all voxels
- select voxels with behaviour/location
- incorporate locality information
  - in voxels considered
  - in classifier

incorporation of domain knowledge and hypotheses to test
what questions can be tackled?

- is there information? (pattern discrimination)
- where/when is information present? (pattern localization)
- how is information encoded? (pattern characterization)
to get started

- MVPA toolbox (in MATLAB)
  http://code.google.com/p/princeton-mvpa-toolbox

- PyMVPA (in Python)
  http://www.pymvpa.org

- support the whole workflow
  - cross-validation, voxel selection, multiple classifiers,…
  - helpful mailing lists (most people are on both)
to get started

- MVPA toolbox (in MATLAB)
  http://code.google.com/p/princeton-mvpa-toolbox
- PyMVPA (in Python)
  http://www.pymvpa.org
- support the whole workflow
  - cross-validation, voxel selection, multiple classifiers,…
  - helpful mailing lists (most people are on both)

- Searchmight (in MATLAB, shameless plug)
  - http://minerva.csbmb.princeton.edu/searchmight
  - special purpose toolbox for information mapping
  - can be used with MVPA toolbox
Thank you!

questions?