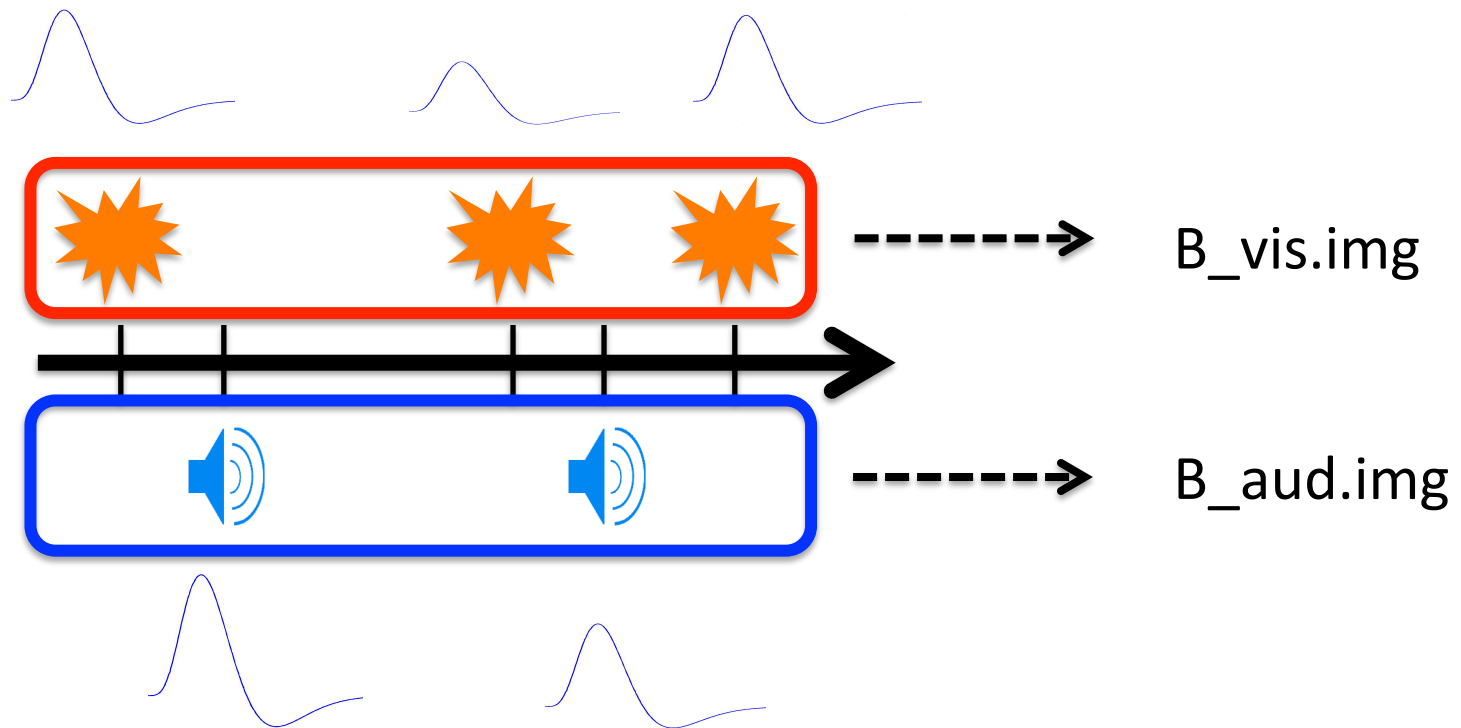


NITP Classification Lab

Wesley Kerr & Don Vaughn

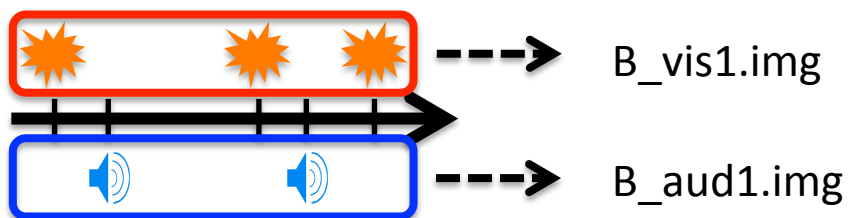


The Paradigm

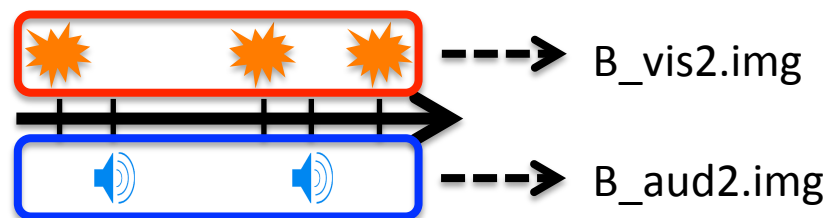


The Model

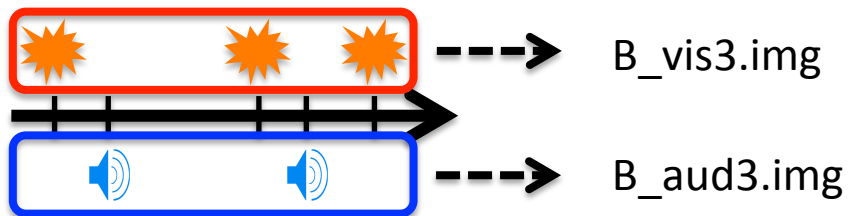
Block 1



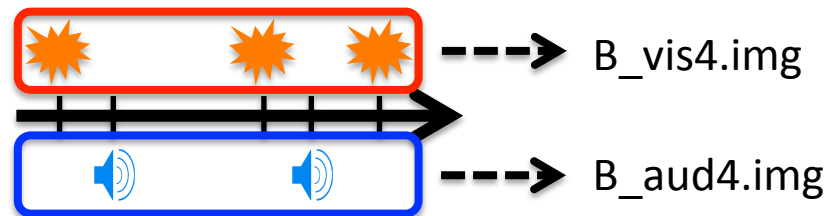
Block 2



Block 3



Block 4



Housekeeping

- Start VirtualBox
- Open Browser and download files
- Open Terminal
 - `cd /usr/local/liblinear/liblinear-1.94/matlab`
 - type: `make`
- Open Matlab, click set path on top right
- Add (w/ subfolders) `/usr/local/spm/spm8`
- Add (w/ subfolders) path to downloaded files
- `addpath /usr/local/liblinear/liblinear-1.94/matlab`

Step1: av03

- `cd *ML_lab/av03/`
- `ls`
 `B_block1_v.img B_block3_v.img B_block6_v.img`
 `B_block1_a.img B_block3_a.img B_block6_a.img`
- `m = d_SVM('_v', '_a')`

Step1: av03

- `cd *ML_lab/av03/`

- `ls`

`B_block1_v.img B_block3_v.img B_block6_v.img`
`B_block1_a.img B_block3_a.img B_block6_a.img`

- `m = d_SVM('_v', '_a')`

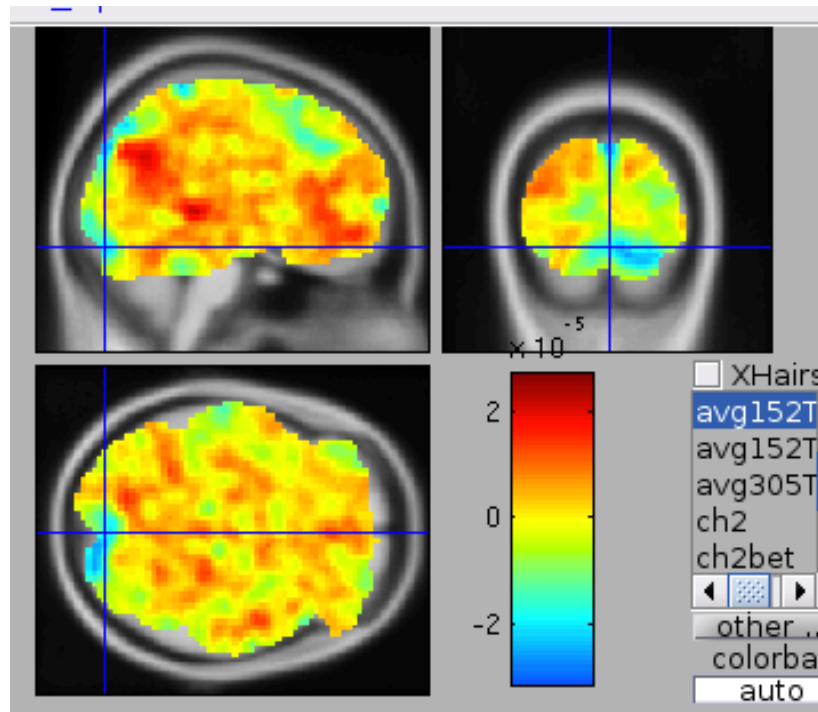
Classification Output

m =

```
      tMat: [4x4 double]
      acc: 1
  accNull: 0.3750
      accP: 0
  weightsP: [53x63x46 double]
          s1: [1x1 struct]
          s2: [1x1 struct]
  accNullDist: [2x1 double]
      weights: [53x63x46 double]
weightsNullDist: [57037x2 double]
  testLabels: [8x1 double]
      preds: [8x1 double]
          DVs: [8x1 double]
```

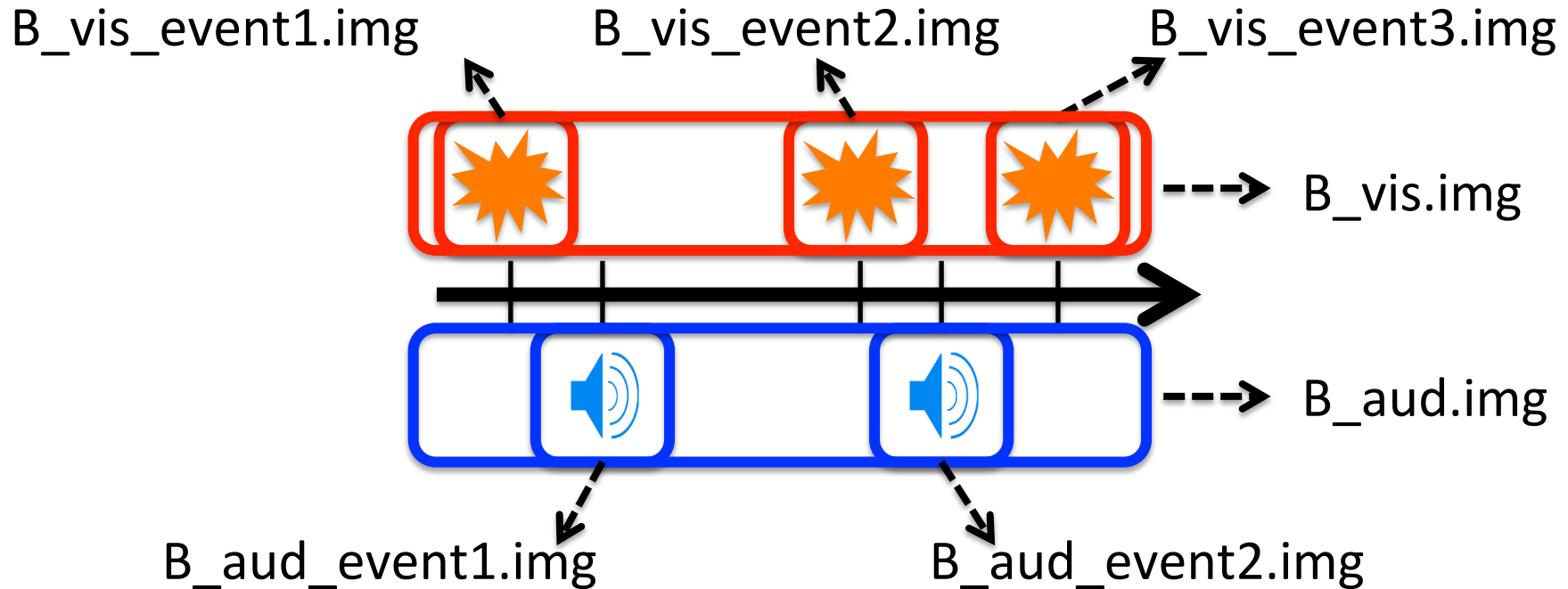
Visualize Your Results

- In Matlab, write these 2 lines:
 - `f_WriteVol('testVol', m.weights, m.tMat)`
 - `xjview`
 - In the xjview window, file -> open testVol.img



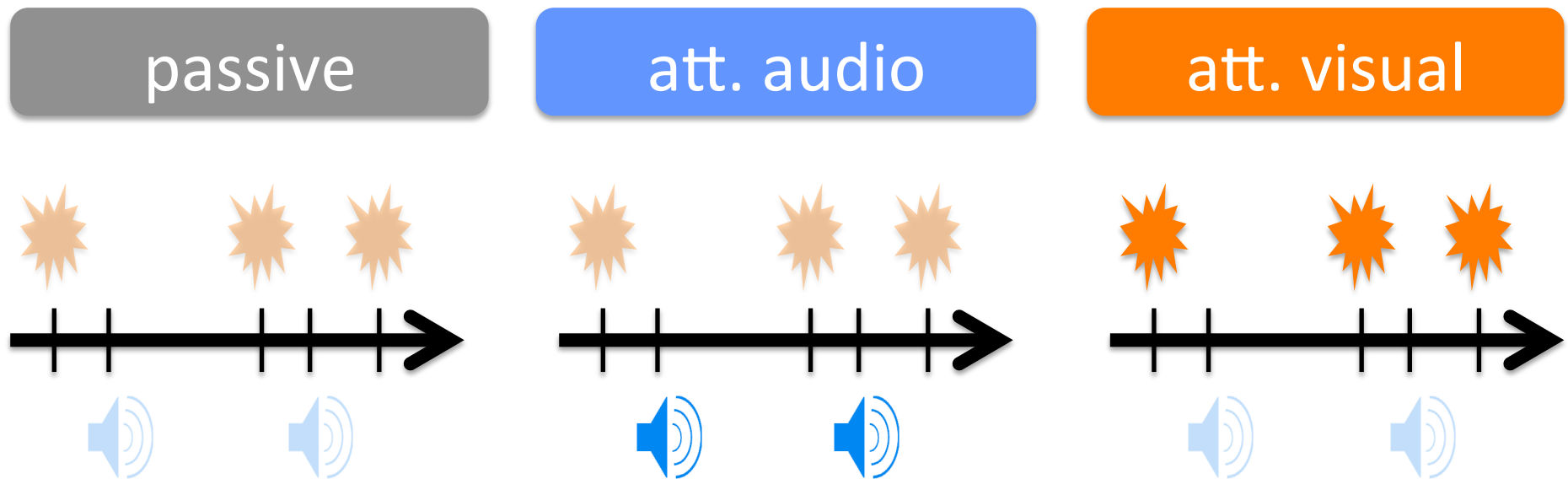
3 Challenges

1: Use Individual Trial Betas



- `cd */ML_lab/av05`
- `ls`
 - `B_block1_a_event1.img B_block1_a_event2.img`
 - `B_block1_v_event1.img B_block1_v_event2.img`
- `m = d_SVM('_a.*.img', '_v.*.img')`

2: Distinguish Attention



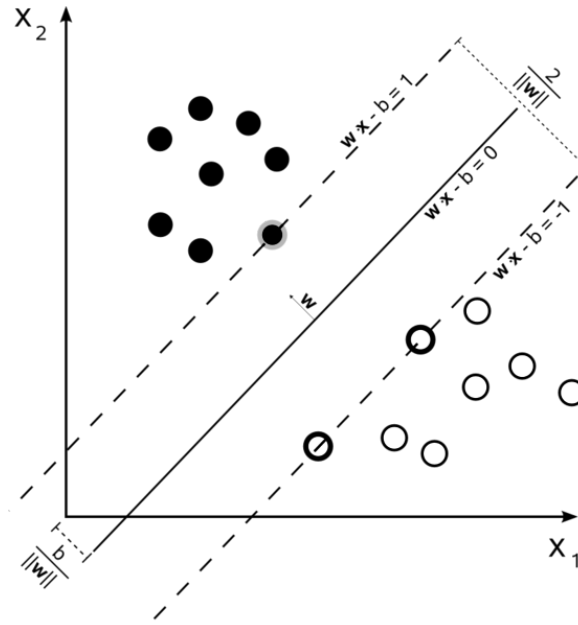
- `cd */ML_lab/av04`
- `ls`
 - `B_block1_audio_a.img` `B_block1_passive_a.img`
 - `B_block1_audio_v.img` `B_block1_visual_a.img`
- `m = d_SVM('visual_a.img', 'audio_a.img')`

attention condition

stimulus type

The diagram shows two dashed arrows. One arrow, labeled 'attention condition', points from the text to the speaker icons in the 'att. audio' and 'att. visual' conditions. The other arrow, labeled 'stimulus type', points from the text to the starburst icons in the 'att. audio' and 'att. visual' conditions.

The SVM model



$$L(\psi|\theta, Y, X) = \frac{\|w\|}{2} + \sum_{i=1}^n \alpha_i [y_i (< w, x_i > + b)] + C \sum_{i=1}^n \xi_i + \sum_{i=1}^n \beta_i \xi_i$$

Regularizer

Linear model

Train documentation

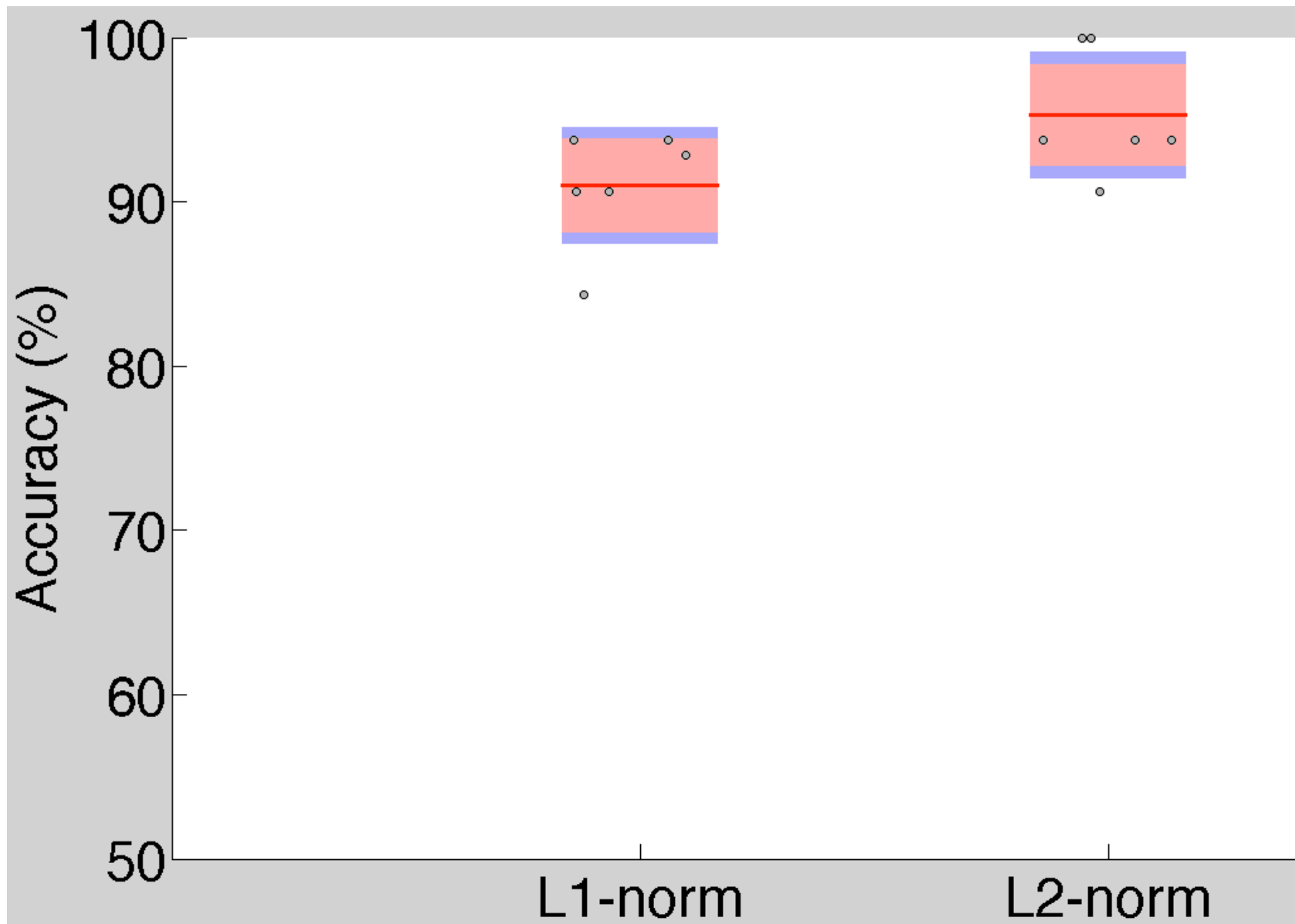
```
liblinear_options:
-s type : set type of solver (default 1)
  for multi-class classification
    0 -- L2-regularized logistic regression (primal)
    1 -- L2-regularized L2-loss support vector classification (dual)
    2 -- L2-regularized L2-loss support vector classification (primal)
    3 -- L2-regularized L1-loss support vector classification (dual)
    4 -- support vector classification by Crammer and Singer
    5 -- L1-regularized L2-loss support vector classification
    6 -- L1-regularized logistic regression
    7 -- L2-regularized logistic regression (dual)

-c cost : set the parameter C (default 1)
-p epsilon : set the epsilon in loss function of SVR (default 0.1)
-e epsilon : set tolerance of termination criterion
```

3: Alter SVM Parameters

- `d_SVM('aud', 'vis', 'nNullPerms', 20)`
 - `'nNullPerms', 20`
 - `'solver', 5` or `'solver', 6` or `'solver', 7`
 - 5: L1-regularized L2-loss
 - 6: L1-regularized logistic regression
 - 7: L2-regularized logistic regression (dual)
 - 1: L2-regularized L2-loss
 - `'eqTrainAmmounts', true`
 - `'testAmmount', 2`

Normalization type impacts accuracy



- 1: Distinguish Attention
 - `cd */ML_lab/av04`
 - `m = d_SVM('visual_a.img','audio_a.img')`
- 2: Use Individual Trial Betas
 - `cd */ML_lab/av05`
 - `m = d_SVM('_a.img','_v.img')`
- 3: Alter SVM Parameters
 - `d_SVM('_a.img','_v.img','nNullPerms', 20)`
 - `'nNullPerms', 20`
 - `'solver', 5` or `'solver', 6` or `'solver', 7`
 - 5: L1-regularized L2-loss
 - 6: L1-regularized logistic regression
 - 7: L2-regularized logistic regression (dual)
 - 1: L2-regularized L2-loss
 - `'eqTrainAmmounts', true`
 - `'testAmmount', 2`

EXTRA, IF TIME

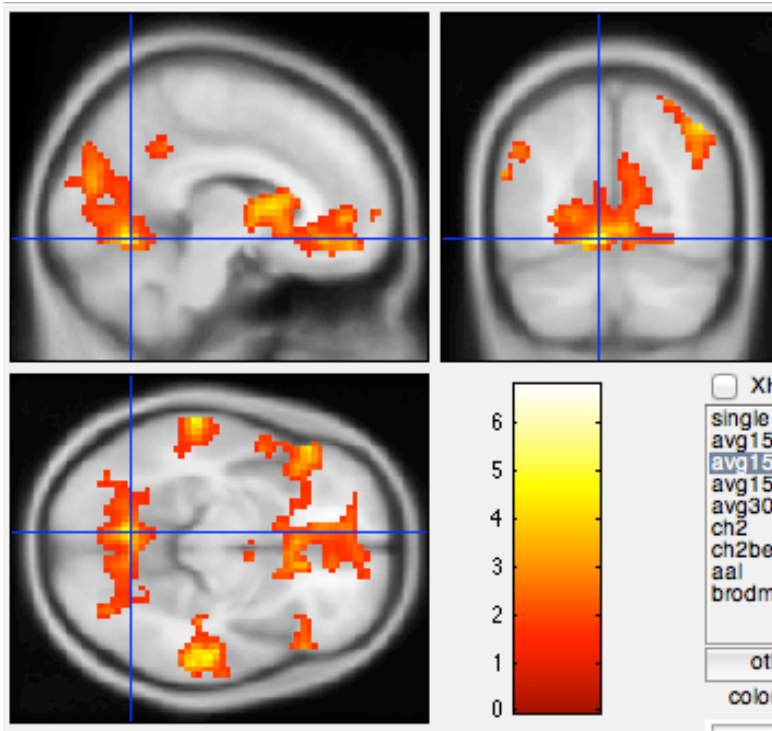
audio



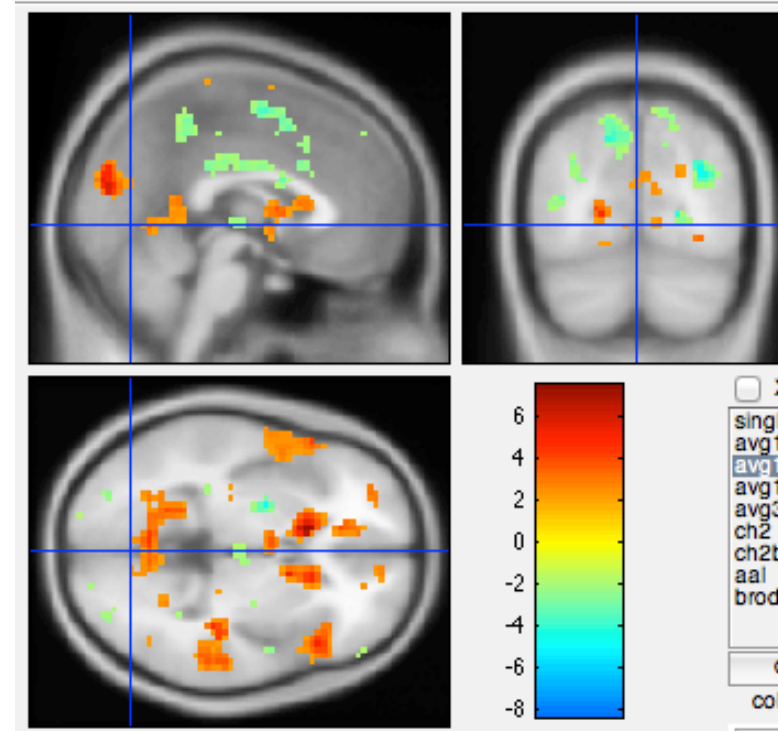
visual



Contrast



L2 Classification



predict documentation

```
Usage: [predicted_label, accuracy, decision_values/prob_estimates] = predict(testing_label_vector, testing_instance_matrix, model, 'liblinear_options',
    [predicted_label] = predict(testing_label_vector, testing_instance_matrix, model, 'liblinear_options','col')
liblinear_options:
-b probability_estimates: whether to output probability estimates, 0 or 1 (default 0); currently for logistic regression only
-q quiet mode (no outputs)
col: if 'col' is setted testing_instance_matrix is parsed in column format, otherwise is in row format
Returns:
predicted_label: prediction output vector.
accuracy: a vector with accuracy, mean squared error, squared correlation coefficient.
prob_estimates: If selected, probability estimate vector.
```