Navigating computational fMRI analysis: a practical guide

[Introduction 2](#_Toc38189350)

[Nancy and Jose 4](#_Toc38189351)

[Questions 5](#_Toc38189352)

[Is the brain sensitive to \_\_\_\_? Does the brain represent \_\_\_\_? 5](#_Toc38189353)

[Univariate analysis 6](#_Toc38189354)

[Multivariate pattern analysis 9](#_Toc38189355)

[Correlation-based MVPA 9](#_Toc38189356)

[Classifier-based MVPA 12](#_Toc38189357)

[Where does the brain represent \_\_\_\_? 16](#_Toc38189358)

[How does the brain represent \_\_\_\_? 16](#_Toc38189359)

[Repetition suppression / adaptation 16](#_Toc38189360)

[Representation similarity analysis / Pattern-similarity MVPA 17](#_Toc38189361)

[How is the brain connected? 18](#_Toc38189362)

[How is the brain’s involvement in \_\_\_\_ shared/different between people? 18](#_Toc38189363)

[List of methods as reference 19](#_Toc38189364)

[Decoding (is there something there) 19](#_Toc38189365)

[Encoding (how is it working) 20](#_Toc38189366)

[Connectivity (how are things working together) 20](#_Toc38189367)

[Other 20](#_Toc38189368)

# 

# Introduction

Functional magnetic resonance imaging (fMRI) analysis can appear to be a daunting and complex task. There are many steps involved to transform raw DICOM files (the standard output format for medical scans) to a preprocessed BOLD[[1]](#footnote-1) time series that is properly corrected for distortion, timing, and motion. This preprocessed time series is then further processed in some way, such as general linear model estimation, to obtain the output necessary (e.g., average activation for each experimental condition for each voxel[[2]](#footnote-2) in a brain region) to finally conduct what we will refer to as “post-processing analyses” (e.g., multivariate pattern analysis). The results of post-processing analyses are leveraged to make conclusions about how the brain works. Although recent tools have greatly decreased the barrier to entry when it comes to fMRI preprocessing (e.g., fMRIPrep) and processing (e.g., Nipype), the various steps required to go from DICOM to post-processing analysis may be one reason for the belief that computational fMRI analysis is difficult and confusing. When a researcher finally reaches the post-processing stage where their data is in its most interpretable state, the researcher might not have easy access to this information (e.g., interpreting the contents of an SPM.mat file is a cruel fate) and only be able to access it via package- and method-specific toolboxes that further obscure our understanding of the steps performed on our data under the hood. These unfortunate realities surrounding fMRI analysis have made the teaching of computational approaches to fMRI needlessly complex.

The goal of this paper is to review the most widely used post-processing fMRI methods, including both basic and advanced computational approaches to post-processing data, in an intuitive and hands-on manner. The intended audience for this paper is researchers who are unsure of which method to use to answer their research question or researchers who might otherwise lack a computational or mathematical background. However, we believe that all neuroimagers may benefit from this condensed review because of the breadth of methods covered and the practicality of open-source code demonstrating how each method can be implemented.

The format of this paper consists of a list of research questions. For each question, we detail how several methods can be used to answer each question. We recommend reading through the research questions in order, however, each section was written such that it could be read independently from the rest of the text. We also follow the stories of two aspiring neuroimagers, Nancy and Jose, as examples of how these methods may be useful for their specific research goals. As Nancy and Jose adopt various methods to answer their research questions, we provide open-source Python code (using simulated fMRI data) containing step-by-step instructions on how to implement each method. The use of simulated data bypasses the fMRI preprocessing and processing steps as well as the complexities surrounding neuroimaging packages and toolboxes. This helps to strip away some of the confusions surrounding fMRI approaches, hopefully demonstrating that most of these methods are fairly straightforward and simple to implement. In the interest of accessibility, we will focus on a high-level understanding of each method and we will not go into specific details that may be necessary to consider when actually using each method on real data (e.g., for classifier-based multivariate pattern analysis, we will describe the basics of support vector machines and why this method has advantages/disadvantages over other methods, but we will not discuss topics such as hyperplane margins, the math underlying support vector machines, or how the preprocessing step of spatial smoothing can affect decoding performance).

## Nancy and Jose

Nancy and Jose are aspiring neuroimagers who are preparing to run an fMRI study. Nancy is a researcher who is interested in how our brains represent spatial location and Jose is a researcher who is interested in how our brains represent movies. As we progress through descriptions of various methods applied to different research questions, Nancy and Jose will pop in to demonstrate how they could use certain methods and what their results would imply about the processing of the human brain. The code Nancy and Jose use to apply these methods to simulated data will be provided via external links to a Python-based Jupyter notebook.

# 

# Questions

## Does the brain represent \_\_\_\_?

The neurons in your brain are consistently firing, and any stimulus will invariably change the rate of firing for at least one neuron out of the approximately 86 billion neurons in the brain. This means that any question along the lines of, “is the brain sensitive to \_\_\_\_\_?” or “does the brain represent \_\_\_\_\_?”[[3]](#footnote-3) will almost assuredly be *yes*. In other words, it is not noteworthy to demonstrate that the brain changes in response to something. It can be noteworthy to show, in specific brain regions, if there is a *difference* in how the brain activates in response to one stimulus compared to another stimulus. The takeaway here is that analyzing within brain regions and using a control condition are essential considerations when conducting fMRI experiments (see “where does the brain represent \_\_\_\_\_?” section for selecting brain regions of interest).

As an example, the research question, “is the brain sensitive to Jennifer Aniston?” is lacking because the brain *will* change in some way if you present someone with a picture of anything (the same is true for any input that gets processed by the brain, not just pictures). What control condition might you use if you want to better investigate this question? Whether your control condition is perceiving nothing, or perceiving strangers faces, or perceiving the face of other celebrities, the more similar the control condition is to your experimental condition, the more your interpretations and conclusions can be fine-tuned. In addition, whether a difference in brain activity between the control condition and your experimental condition is observed in the primary visual cortex (possible difference in early visual processing) or the amygdala (possible difference in emotional processing) can suggest different interpretations, given what is already understood about certain brain regions.

|  |  |  |
| --- | --- | --- |
| Univariate analysis | Correlation-based MVPA | Classifier-based MVPA |
| Difference in averaged activations is not equal to zero | Stronger linear relationship for within-condition data than between-condition data | (Linearly) decodable difference in brain activations between conditions | |

Table 1. Overview of the methods we will discuss in this section that can be used to determine whether a brain region of interest processed conditions differently across subjects. Note that not all methods will converge to the same result as these methods involve statistical tests targeting different questions.

### Univariate analysis

Now that we have a well-defined research question in terms of comparing brain activation between conditions, there are several approaches to fMRI analysis that are available. The most straightforward approach is univariate analysis (see Figure 1 to see how Jose used a univariate analysis to test whether a brain region processed horror movies differently than comedy movies). For a univariate analysis, you need, for each participant, the average brain activation for one condition and the average brain activation for another condition (beta weights are often used as the estimate of brain activity, where beta weights are obtained from general linear model estimation[[4]](#footnote-4)) within a brain region. Then you can simply conduct a t-test across participants on the difference in brain activations between conditions. This allows you to test whether a brain region is consistently more or less active across people in response to certain conditions. Importantly, you cannot determine *why* there is a difference between conditions based on a univariate analysis (see “how does the brain represent \_\_\_\_\_?” section).

**Univariate Analysis**

Does this brain region process horror movies differently than comedy movies?

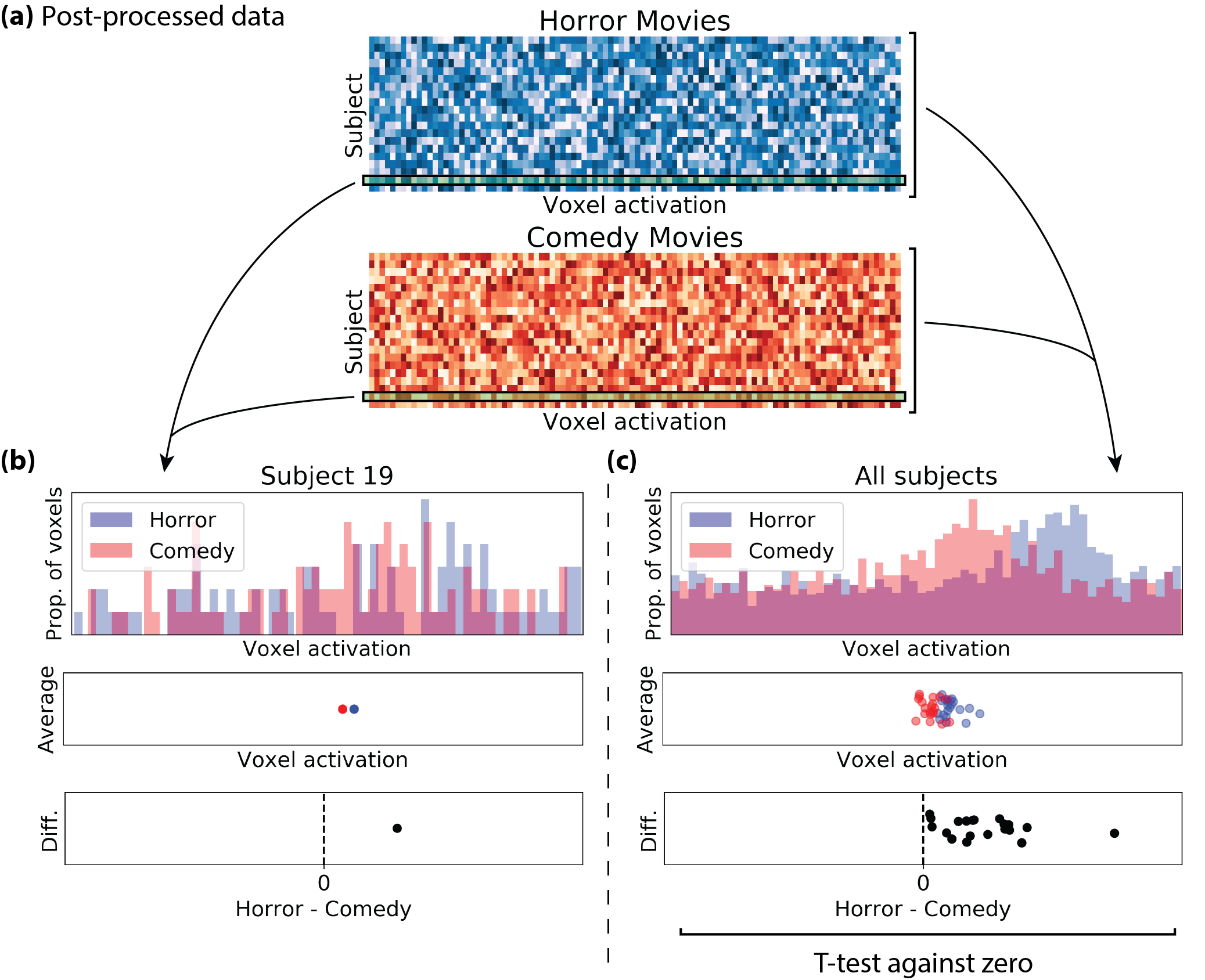


Figure . Example of using a univariate analysis to test if neural activity was different between two conditions (watching a horror movie vs. watching a comedy movie) within a brain region. (a) Post-processed data, visualized as a subject by voxel matrix for each movie condition. Each row represents a different subject and each column represents a different voxel. Activation values are in arbitrary units, with darker colors representing larger activations. (b) Distribution of voxel activations for one subject, followed by plots showing the average activation per condition and the difference of these averaged activations. (c) Repeating the process shown in (b) for all subjects. For each subject, the average voxel activation for the horror condition was subtracted by the average voxel activation for the comedy condition. A positive difference indicates stronger activation for horror and negative difference indicates stronger activation for comedy. A one-sample t-test against zero can be conducted on these difference values to test whether there is a significant difference in how the brain region processed the two task conditions. Y-axis for scatterplots are uninformative: data points were jittered so that data points did not overlap. The Python code used to simulate and analyse this data is available in univariate.ipynb.

### Multivariate pattern analysis

Multivariate (or multivoxel) pattern analysis (MVPA) is another approach that can accomplish similar goals as univariate analysis. MVPA is distinct from univariate analysis because brain activation is not averaged for each condition. This can potentially increase the sensitivity of your data because voxels may carry information that is lost when averaged together with other voxels. When neuroimagers use the term “patterns of neural activity”, they simply mean that activations have not been averaged together. MVPA can refer to a few different analysis approaches, which can roughly be broken down into correlation-based MVPA, classifier-based MVPA, and representational similarity analysis (Haxby, 2001; Lewis-Peacock & Norman, 2013). It is somewhat confusing terminology, because you might think that MVPA covers all neuroimaging analyses where the data is not averaged across conditions, but in practice it typically only refers to a few specific methods. In correlation-based MVPA, voxel activations in a subset of the data are correlated with voxel activations in the held-out data of the same or different task condition. In classifier-based MVPA, a subset of labeled data is used to train a model that can predict the labels of held-out data. Representational similarity analysis (RSA), also known as pattern-similarity MVPA, will be discussed in the “how does the brain represent \_\_\_\_?” section and involves voxel activations being projected into a high-dimensional space where the distance between data points indicates the similarity of brain patterns.

#### Correlation-based MVPA

The simplest kind of MVPA is a correlation-based approach. To illustrate this approach with an example (see Figure 2), Jose’s task was to show participants short clips of movies on every trial across ten fMRI runs[[5]](#footnote-5). Every block of trials was composed of 5-second clips from a horror movie or a comedy movie. To test for a difference in brain activation when watching horror or comedy movies, Jose correlated each voxel’s activation during odd runs to each voxel’s corresponding activation during even runs. Separate correlations were performed for horror blocks and comedy blocks, producing four correlations per voxel (odd-horror🡪even-horror, odd-comedy🡪even-comedy, odd-horror🡪even-comedy, and odd-comedy🡪even-horror). Jose found significantly stronger correlations between blocks of the same movie genre compared to correlations between blocks of different movie genres, suggesting that this brain region processed the horror movie clips differently than the comedy movie clips.

**Correlation-based MVPA**

Does this brain region process horror movies differently than comedy movies?

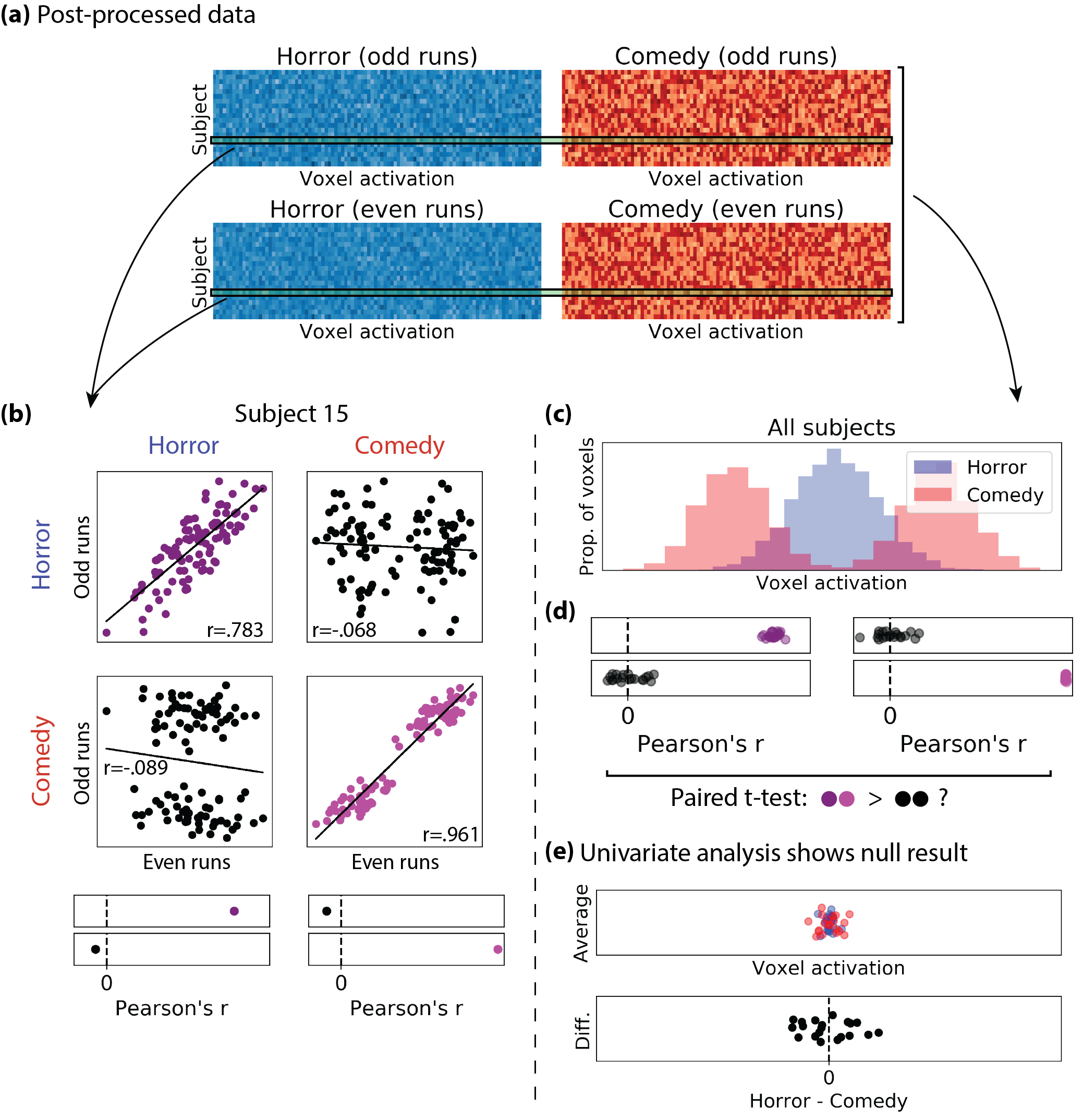
****

Figure 2. Example of correlation-based MVPA to test if neural activity was different between two conditions (watching a horror movie vs. watching a comedy movie) within a brain region. (a) Post-processed data, visualized for even and odd runs as a subject by voxel matrix for each movie condition. (d) For each subject, voxel activations were correlated between odd and even runs, separately for each condition combination: odd-horror🡪even-horror, odd-comedy🡪even-comedy, odd-horror🡪even-comedy, and odd-comedy🡪even-horror. This results in 4 Pearson’s r values per subject. (c) Distribution of voxel activations, collapsed across subjects, for each condition. (d) Across subjects, a paired t-test between correlations of the same condition (horror🡪horror and comedy🡪comedy) and correlations of different conditions (horror🡪comedy and comedy🡪horror) tests for a difference in how the brain region processed the two task conditions. (e) A univariate analysis would not work with this example dataset because of how voxel activations were averaged per condition (i.e., in panel c, the average of the two red bumps is the same as the average of the one blue bump). The Python code used to simulate and analyse Jose’s data is available in mvpa\_nonlinear.ipynb.

#### Classifier-based MVPA

An important distinction between the above correlation-based approach and a classifier-based approach is that the former is statistical modeling and the latter is machine learning. That is, correlation-based MVPA fits your data to a simple statistical model which yields a strength of association (e.g., the standardized slope of a linear regression between odd runs and even runs). Correlation-based MVPA does not make predictions about whether a given input is a horror movie or a comedy movie. Classifier-based MVPA, on the other hand, does make such predictions. Predicting the correct label above chance on held-out data is the test used to support whether a brain region processed conditions differently. Classifier-based MVPA is also different from correlation-based MVPA because (1) it may require more data because some data will need to be held-out to test the model, (2) it can weigh certain voxels over other voxels to maximize the signal-to-noise ratio of your data, and (3) depending on the classifier and cross-validation[[6]](#footnote-6) strategy used, classifier-based approaches can be prone to overfitting. Also note that in machine learning language, voxels are “features” and task conditions are “classes”.

Using the same as example of horror movies and comedy movies, Jose can implement a classifier-based MVPA approach by using data from odd runs as the training data and data from even runs as his test data. First imagine that Jose has only two voxels in his brain region of interest (Figure 3). Using his training data, he can display a scatterplot where the y-axis is voxel 1’s activations and the x-axis is voxel 2’s activations, and each data point is colored according to whether the block was a horror movie or a comedy movie. Here, the x-axis and the y-axis are the two “dimensions” of the dataset. The “model” (aka classifier, decoder, or support vector machine) uses this training data to fit a decision boundary (aka hyperplane in the case of several dimensions) that best separates the data points according to their classification (horror or comedy). This decision boundary is then used to classify the labels of the test data. In this example of two dimensions, a human could reasonably estimate the placement of the decision boundary without the need for a model. However, when using a region of interest with more than two voxels, the number of dimensions increases such that a simple scatterplot is no longer possible.

A close up of a map

Description automatically generated

Figure 3. Simplified example of a support vector machine, using a two-voxel region of interest. Training data are used to fit a decision boundary, or hyperplane, which can best separate the two classes. This decision boundary is then used to separate the classes of the test data. Dots in the red shaded region are predicted to be red dots, and dots in the blue shaded region are predicted to be blue dots. The number of dimensions is equal to the number of voxels, making this approach difficult to visualize with a large number of voxels.

In the more detailed example shown in Figure 4, Jose had a 100-voxel region of interest where voxel activations were averaged such that there were four activation values per voxel, split by odd/even run and horror/comedy condition. For every subject, a multidimensional decision boundary was fit to the training data (odd runs) to best separate the horror and comedy conditions. This decision boundary was then used to predict the classes of the test data (even runs). Average accuracy was computed, and Jose found that accuracies across subjects was significantly above chance (50% accuracy, since there are two classes), suggesting that this brain region processed the horror movie clips differently than the comedy movie clips.

**Classifier-based MVPA**

Does this brain region process horror movies differently than comedy movies?

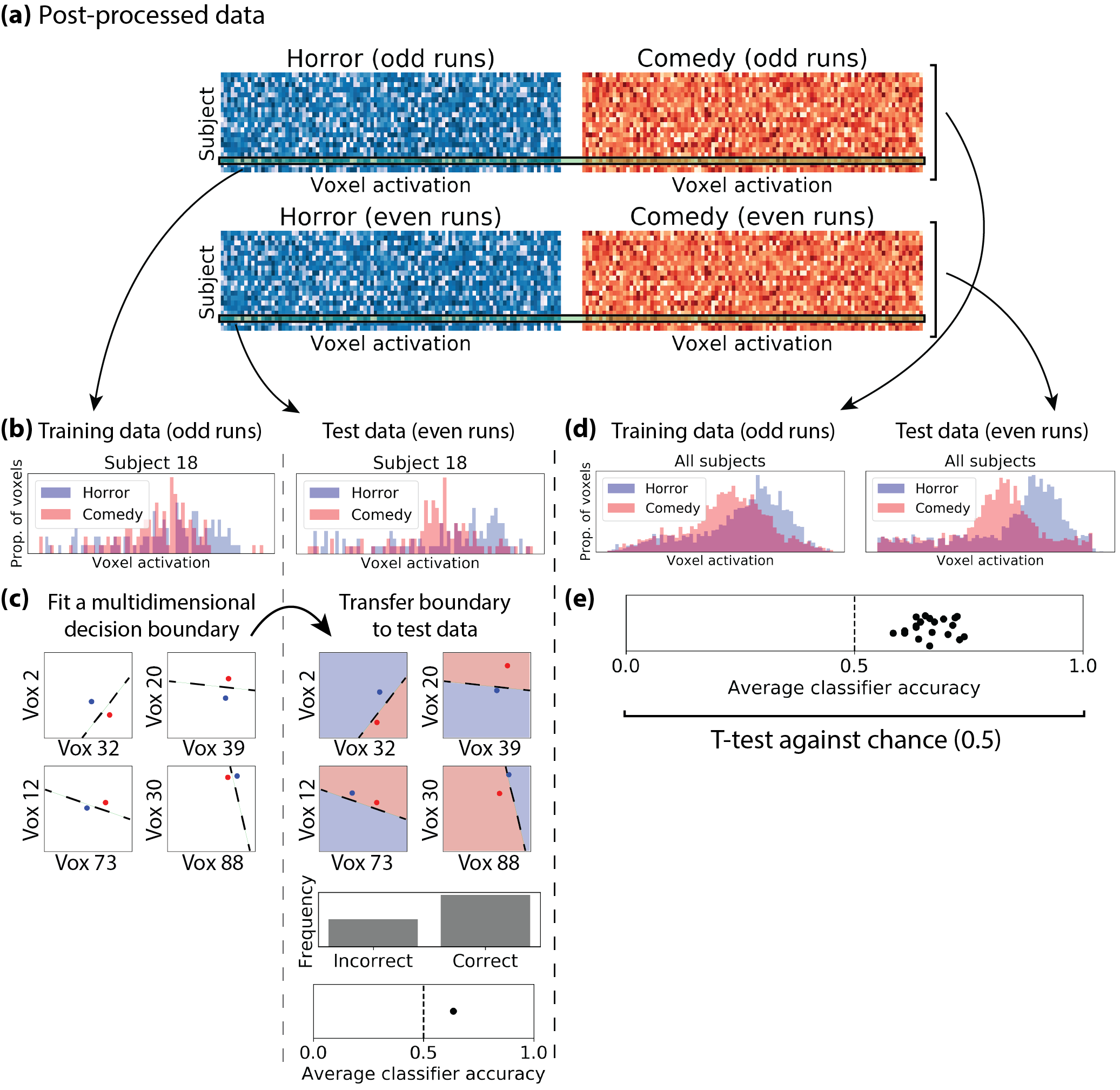


Figure 4. Example of classifier-based MVPA to test if neural activity was different between two conditions (watching a horror movie vs. watching a comedy movie) within a brain region. (a) Post-processed data, visualized for even and odd runs as a subject by voxel matrix for each movie condition. (b) Distribution of voxel activations for training data (odd runs) and test data (even runs) for one subject. (c) For each subject, a multidimensional decision boundary (aka hyperplane) is fit across features (voxels) best separate classes (conditions). This decision boundary is then transferred to the held-out, test data. This boundary predicts the classes of test data, yielding an average classifier accuracy score per subject. (d) Distribution of voxel activations, collapsed across subjects, for training data and test data. (e) Across subjects, a t-test against chance classifier accuracy (50% for two classes) tests for a difference in how the brain region processed the two task conditions. The Python code used to simulate and analyse Jose’s data is available in mvpa\_linear.ipynb.

## Where does the brain represent \_\_\_\_?

T-maps, spotlight analysis, phase-encoded mapping, population receptive field mapping

## How does the brain represent \_\_\_\_?

#### Repetition suppression / adaptation

Repetition suppression is a kind of univariate analysis to test whether a brain region of interest responds differently between conditions. The method relies on the fact that single-cell recordings show that neurons evoke a relatively suppressed response to the repeated presentation of its preferred stimulus (REF). For example, a neuron that is sensitive to horizontally oriented lines might show a very strong response to the presentation of a horizontal line but only a moderately large response to the horizontal line when re-presented a few seconds later. In other words, repetition suppression refers to the observation that a region of interest that is selective to stimulus X will show decreased activation to stimulus X preceded by stimulus X compared to stimulus X preceded by stimulus not-X. Repetition suppression is an extension of univariate analysis that requires an experimental design where conditions are sometimes repeatedly presented.

Repetition suppression is usually used to define a region of interest’s range of stimulus selectivity. To illustrate this with a real-world example of repetition suppression, Kourtzi & Kanwisher (2001) were interested in the selectivity of a region of object-selective cortex called the lateral occipital cortex (LOC). In one experiment they had four conditions: (1) object preceded by same object, (2) object preceded by the same object with a different location in depth, (3) object preceded by the same object with a different shape, and (4) object preceded by a completely different object (change in shape and depth). Only conditions 1 and 2 showed repetition suppression, meaning that an object perceived as close to you was represented similarly to the same object perceived far away from you. In other words, LOC was invariant to changes in location in depth.

Grill-Spector et al. (1999) were interested in whether a region of object-selective cortex called the lateral occipital cortex (LOC) would show repetition suppression for objects shown in different viewing conditions: size, position, illumination, and viewpoint.

If a brain region contains neurons that are sensitive to pictures of houses, for instance, then the brain region should show an overall decreased activation to the repeated presentation of a house. If the brain region is not sensitive to pictures of houses, then there should be no change in activation to the repeated presentation of a house. The analysis steps involved in repetition suppression are identical to the aforementioned steps in univariate analysis except that the conditions are now whether the stimulus was repeated or not.

----

The use of the term “representation” in neuroscience generally refers to a systematic relationship between features of the natural world and the activity of

neurons in the brain.

----

#### Representation similarity analysis / Pattern-similarity MVPA

Representation similarity analysis (RSA, aka Pattern-similarity MVPA) is used to relate brain activity between modalities (Kriegeskorte, Mur, & Bandettini, 2008).

## How is the brain connected?

Lorem

## How is the brain’s involvement in \_\_\_\_ shared/different between people?

Shared response modeling, clinical populations, …

# 

# List of methods as reference

(roughly categorized -- this categorization will not be in the paper, I just thought repeating our list of methods might be helpful)

Repetition suppression

<https://journals.physiology.org/doi/full/10.1152/jn.90376.2008>

<https://royalsocietypublishing.org/doi/10.1098/rstb.2015.0355#d3e698>

<https://www.sciencedirect.com/science/article/pii/S0896627300808326#FIG4>

<https://science.sciencemag.org/content/293/5534/1506?casa_token=VBNXICeEm0UAAAAA:7LDr-zZvbGWU_Nd1Em3fRgKAm-NeHZbEhDdOsYJWnWQVquch9wgVjJMrOMElcZhiuDgns6GOU3FDNA>

## Decoding (is there something there)

Standard GLM, repetition suppression,

mvpa

<https://www.annualreviews.org/doi/pdf/10.1146/annurev-neuro-062012-170325>

rsa

Haxby *et al.,* 2014; Kriegeskorte *et al.,* 2008a

<https://academic.oup.com/scan/article/14/11/1243/5693905>

iem

cross-subject decoding

## Encoding (how is it working)

Phase-encoded mapping, population receptive field mapping, IIEM (refer to preprint), PCM?, neural network (see Princeton neuromatch poster)

## Connectivity (how are things working together)

Functional connectivity (task-based, resting-state (see Noble et al., 2017), background), beta series (related to task-based connectivity) (rissman et al 2004), informational connectivity, connectome, connectivity fingerprinting, psychophysiological interaction (PPI; aka context-dependent correlations in AFNI), graph theory (ROI or voxels as nodes, correlation as edges, number/proportion of edges for each node), dynamic causal modeling (i.e., effective connectivity), shared response modeling

## Other

Whole brain / ROI, HRF / FIR, Block / Event, incorporating behavioral measures, neurofeedback / time-scale, pre-analysis considerations, preprocessing

1. BOLD stands for blood-oxygen-level-dependent, referring to how fMRI measures changes in blood oxygenation as a proxy for neuronal activity. [↑](#footnote-ref-1)
2. MRI scans are built up of 3D pixels called voxels, the portmanteau of “volume” and “pixel”. Voxels can vary in size depending on the scanning protocol, but typically a voxel contains around a million neurons. [↑](#footnote-ref-2)
3. It could be argued that these two questions are fundamentally different, such that “representation” means that researchers are testing for a systematic relationship between the stimulus and neural activity. This will be discussed in more depth in the “how does the brain represent \_\_\_\_?” section. [↑](#footnote-ref-3)
4. Regarding general linear models, “general” refers to the different types of analyses that can be used (e.g., correlation, one-sample t-test, ANOVA, etc.) and “linear” refers to how the model assumes a linear relationship between parameters (often preferred because nonlinear models are prone to overfitting, where a model appears to fit the data well but is not generalizable to new data). We consider the general linear model to be a processing step and thus will not discuss it in further detail (see REF for review), but it is essentially one way to obtain an estimate of brain activation (in arbitrary units) for every voxel for each condition after accounting for things like the hemodynamic response function (time-course of blood flow) and body motion in the scanner. [↑](#footnote-ref-4)
5. Typically, an fMRI session is divided into several 5-10 minute “runs”, where each run consists of a single task and at the end of every run the participant has a chance for a short break. Depending on the design, runs can be further subdivided into “blocks”, or a series of trials of the same task condition. [↑](#footnote-ref-5)
6. Cross-validation refers to how a model’s performance is validated by making predicting about data that was not used to train the model. Here we will use a simple cross-validation approach: using odd runs as training data and even runs as test data (aka half of a 2-fold cross-validation). [↑](#footnote-ref-6)