

neuro_decode_practice

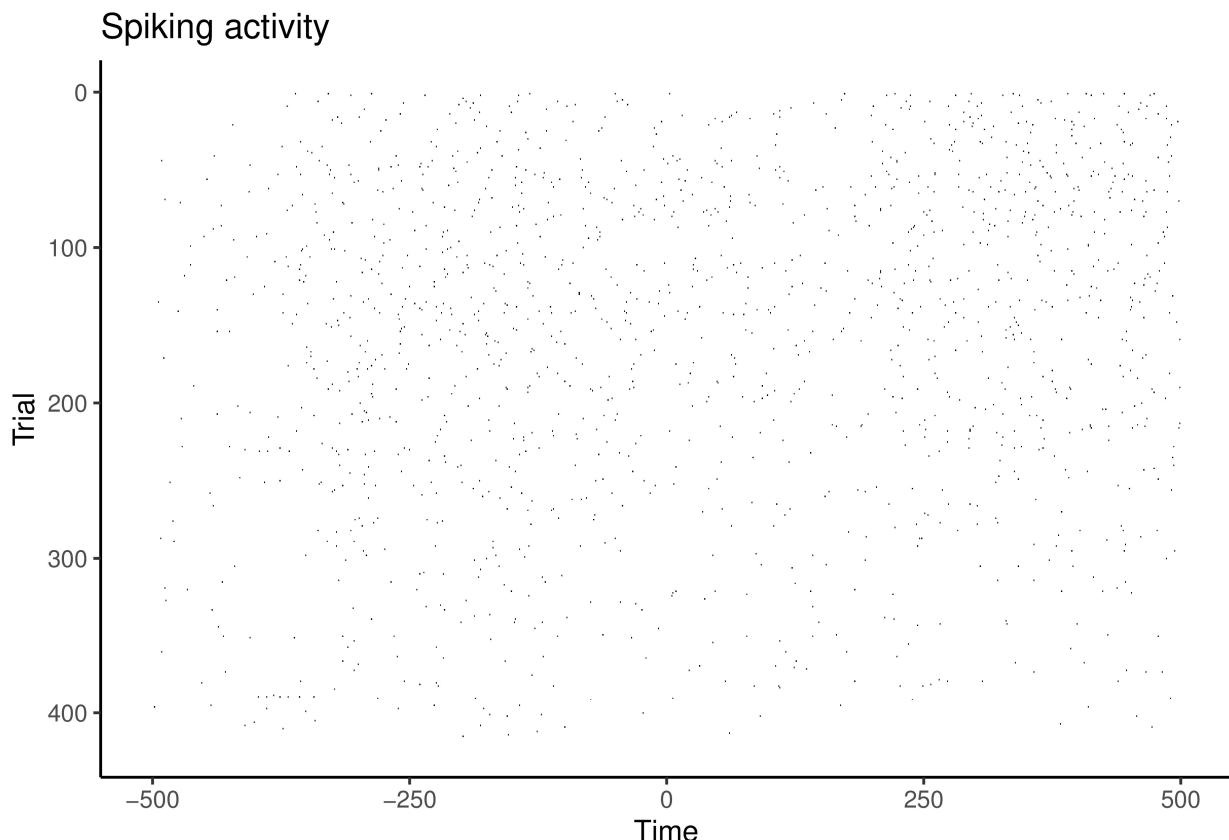
Brennan

2022-11-07

```
# Load in the data in the desired format
raster_dir_name <- file.path(system.file("extdata", package = "NeuroDecodeR"),
                             "Zhang_Desimone_7object_raster_data_small_rda")
file_name <- "bp1001spk_01A_raster_data.rda"
load(file.path(raster_dir_name, file_name))

test_valid_raster_format(file.path(raster_dir_name, file_name))

# Graph the data as a sanity check
plot(raster_data)
```



```
# Group the data into bins
save_dir_name <- file.path("Test_Binned")
```

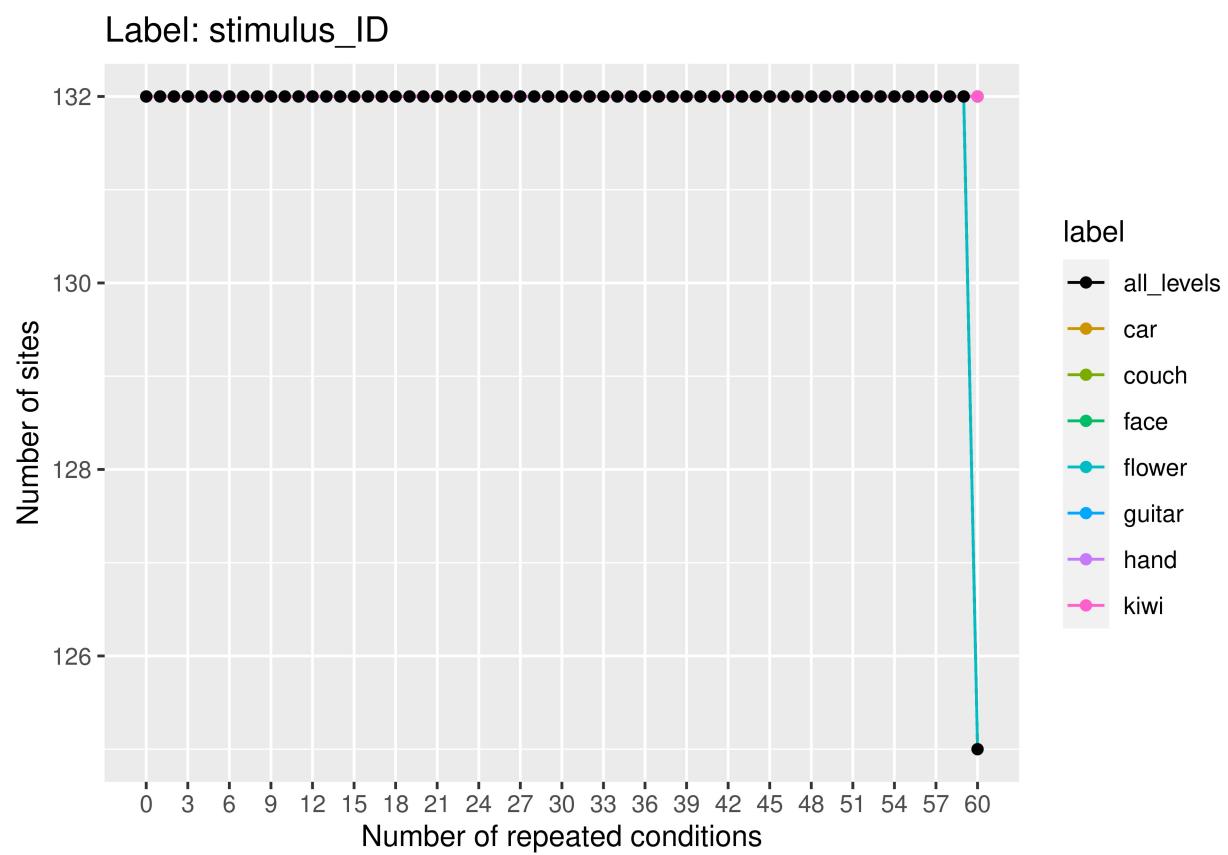
```

binned_file_name <- create_binned_data(raster_dir_name, file.path(save_dir_name, "ZD"),
                                         150, 50, num_parallel_cores = 2)

## | 

# Visualize how many times each stimulus was presented
binned_file_name <- system.file(file.path("extdata", "ZD_150bins_50sampled.Rda"), package="NeuroDecodeR")
label_rep_info <- get_num_label_repetitions(binned_file_name, "stimulus_ID")
plot(label_rep_info)

```



Decoding Analysis

Performing a decoding analysis involves several steps:

1. Creating a datasource (DS) object that generates training and test splits of the data.

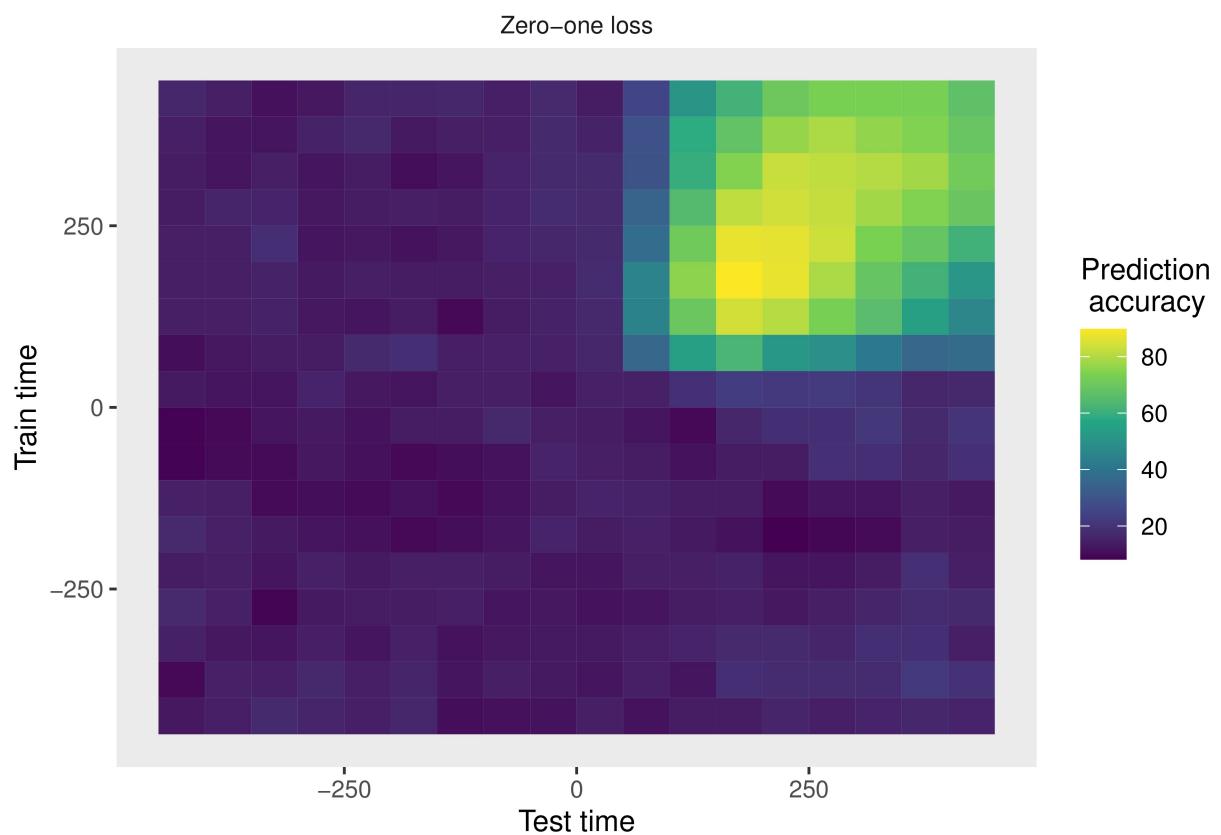
```

# Get the data
binned_file_name <- system.file(file.path("extdata", "ZD_150bins_50sampled.Rda"),
                                  package="NeuroDecodeR")

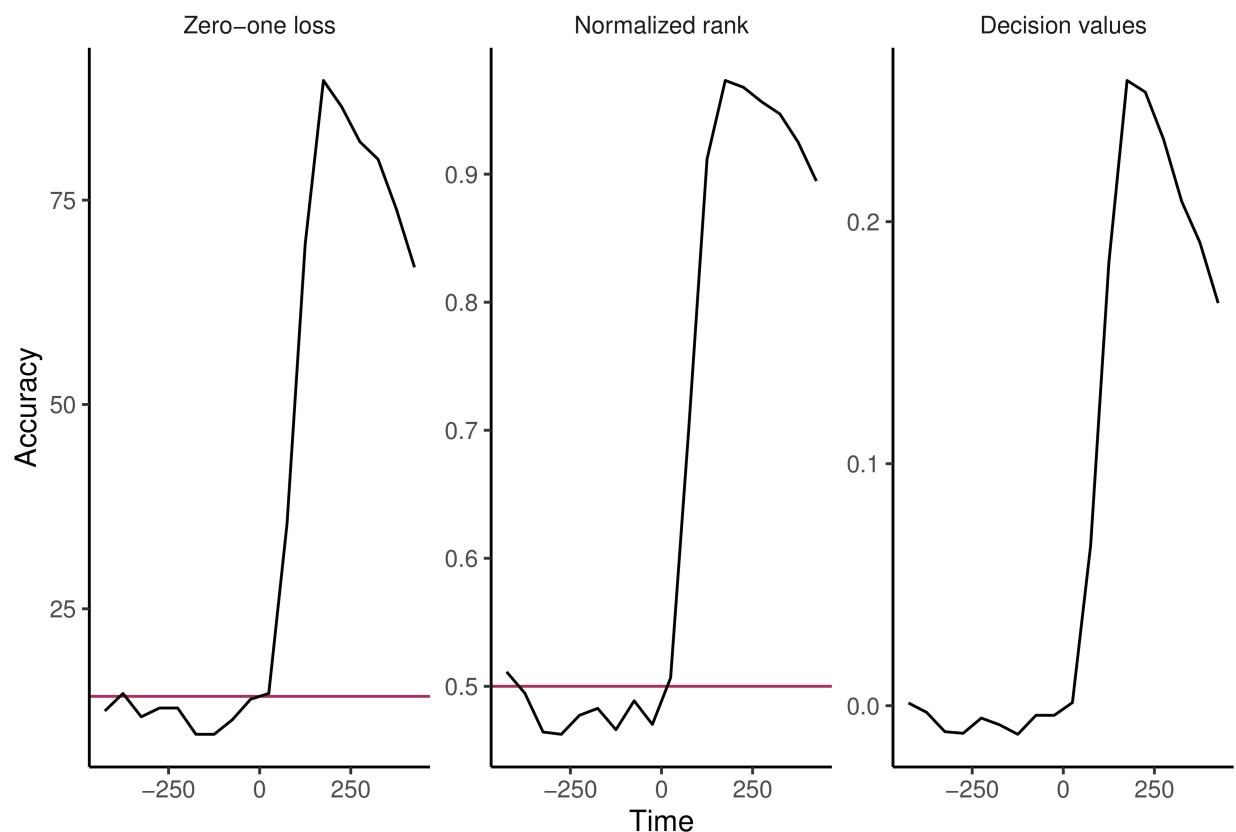
# Get the response variable
variable_to_decode <- "stimulus_ID"

# Determine the number of splits (this is lower than maximum in this case for runtime)

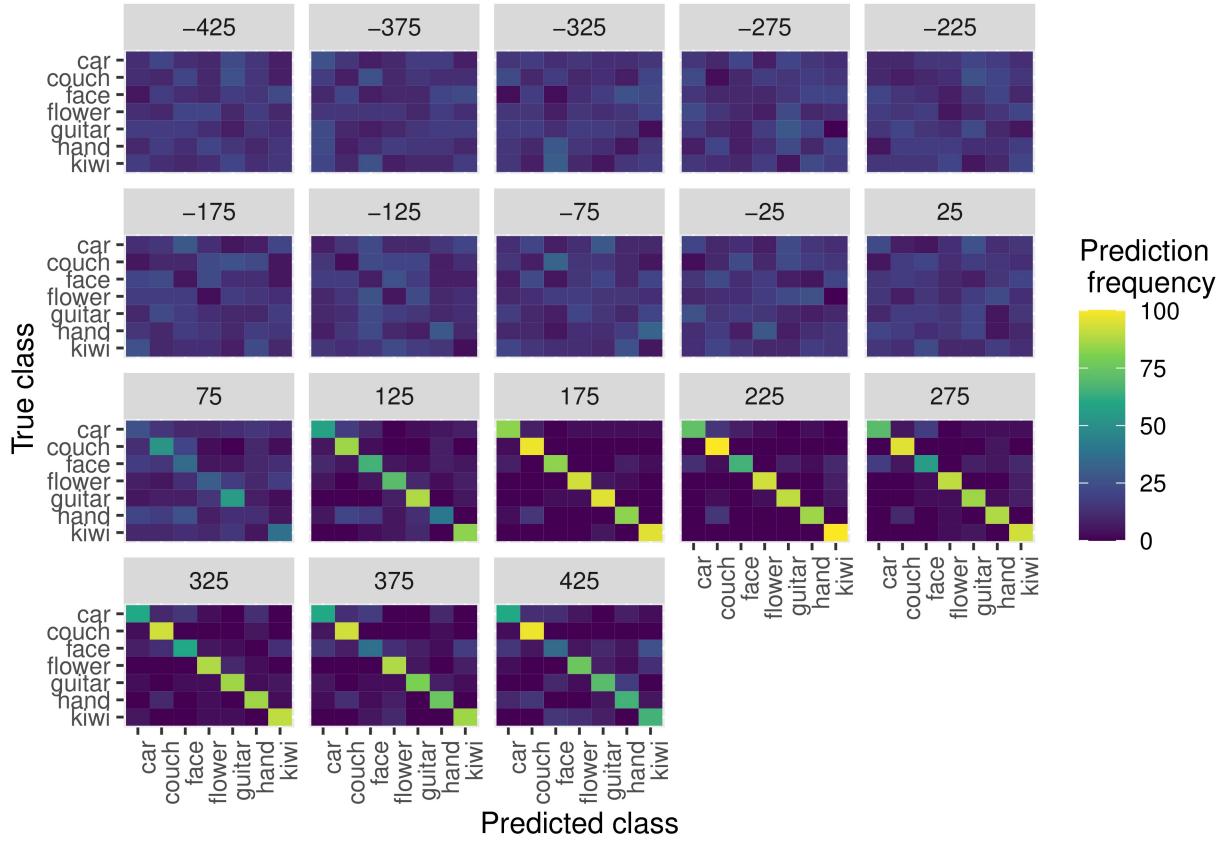
```

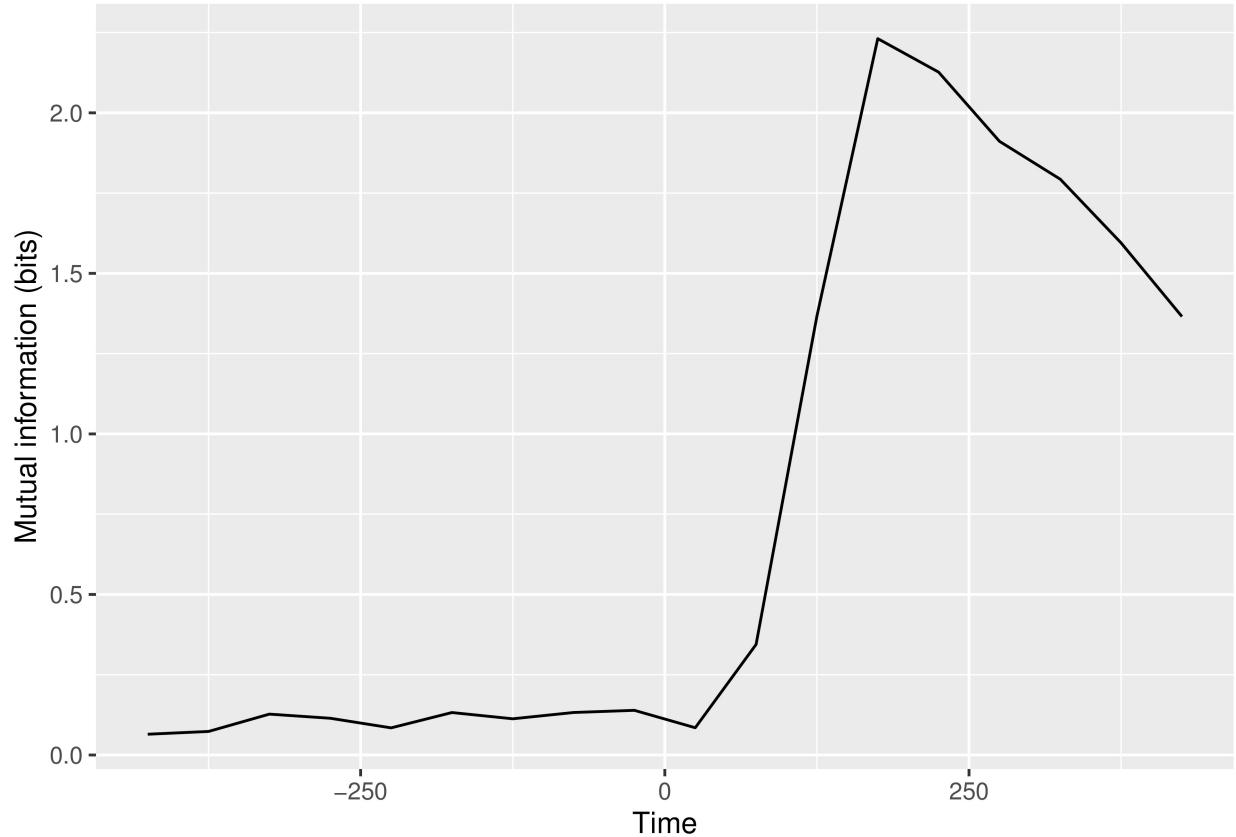
```
plot(DECODING_RESULTS$rm_main_results, results_to_show = 'all', type = 'line')
```



```
# Plot the confusion matrix
plot(DECODING_RESULTS$rm_confusion_matrix)
```



```
plot(DECODING_RESULTS$rm_confusion_matrix, results_to_show = "mutual_information")
```



```
# This code saves the data to a file if there is a valid directory listed
results_dir_name <- file.path(tempdir(), "results", "")
dir.create(results_dir_name)

log_save_results(DECODING_RESULTS, results_dir_name)

## Warning in log_save_results(DECODING_RESULTS, results_dir_name): The manifest file does not exist.
## Assuming this is the first result that is saved and creating manifest file
```