```
!wget -c https://uni-bonn.sciebo.de/s/f81JqGZmEHXxMnB/download?path=%2F&files=steinmetz_2016-12-14_Cori.nc
```

```
--2023-10-24 12:53:34--  https://uni-bonn.sciebo.de/s/f81JqGZmEHXxMnB/download?path=%2F
Resolving uni-bonn.sciebo.de (uni-bonn.sciebo.de)... 128.176.1.2
Connecting to uni-bonn.sciebo.de (uni-bonn.sciebo.de)|128.176.1.2|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: ‘download?path=%2F’

download?path=%2F       [      <=>           ]   1.16G  27.1MB/s    in 43s

2023-10-24 12:54:18 (27.6 MB/s) - ‘download?path=%2F’ saved [1241807443]
```

```
!unzip /content/download?path=%2F
```

```
Archive:  /content/download?path=%2F
   creating: steinmetz_neuromatch_dataset/
 extracting: steinmetz_neuromatch_dataset/steinmetz_2016-12-14_Cori.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2016-12-17_Cori.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2016-12-18_Cori.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-07_Muller.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-08_Muller.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-08_Radnitz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-09_Muller.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-09_Radnitz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-10_Radnitz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-11_Radnitz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-01-12_Radnitz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-05-15_Moniz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-05-16_Moniz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-05-18_Moniz.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-06-15_Hench.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-06-16_Hench.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-06-17_Hench.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-06-18_Hench.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-10-11_Theiler.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-10-29_Richards.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-10-30_Richards.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-10-31_Richards.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-01_Forssmann.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-01_Richards.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-02_Forssmann.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-02_Richards.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-04_Forssmann.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-11-05_Forssmann.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-05_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-06_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-06_Tatum.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-07_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-07_Tatum.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-08_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-08_Tatum.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-09_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-09_Tatum.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-10_Lederberg.nc
 extracting: steinmetz_neuromatch_dataset/steinmetz_2017-12-11_Lederberg.nc
```

```
import xarray as xa
```

```
dataset = xa.load_dataset('/content/steinmetz_neuromatch_dataset/steinmetz_2016-12-14_Cori.nc')
```

```
dataset
```

```
xarray.Dataset
▶ Dimensions:       (trial: 364, time: 250, cell: 734, waveform_component: 3, sample: 82, probe: 384,
                    brain_area_lfp: 7, spike_id: 2446173)
▼ Coordinates:
    trial           (trial)                 int32   1 2 3 4 5 6 ... 360 361 362 363 364
    time            (time)                float64   0.01 0.02 0.03 ... 2.48 2.49 2.5
    cell            (cell)                  int32   1 2 3 4 5 6 ... 730 731 732 733 734
    waveform_co...  (waveform_component)    int32   1 2 3
    probe           (probe)                 int32   1 2 3 4 5 6 ... 380 381 382 383 384
    brain_area_lfp  (brain_area_lfp)       object   'ACA' 'LS' 'MOs' ... 'SUB' 'VISp'
    spike_id        (spike_id)              int32   1 2 3 4 ... 2446171 2446172 2446173
▶ Data variables:   (31)
▶ Indexes:   (7)
▼ Attributes:
    session_date :  2016-12-14
    mouse :         Cori
    stim_onset :    0.5
    bin_size :      0.01
```

```
dataset.wheel.values
```

```
array([[-1,  0,  0, ...,  1,  0,  1],
       [ 0, -1,  0, ...,  1,  0,  0],
       [ 0,  0, -1, ..., -1,  0,  0],
       ...,
       [ 0,  0,  0, ...,  0,  0,  0],
       [ 0,  0,  0, ...,  0,  0,  0],
       [ 0,  0,  0, ...,  0,  0,  0]], dtype=int8)
```

```
dataset.wheel.to_dataframe()
```

		wheel
trial	**time**	
1	**0.01**	-1

```
wheel_data = dataset.wheel.to_dataframe().reset_index()
wheel_data
```

	trial	time	wheel
0	1	0.01	-1
1	1	0.02	0
2	1	0.03	0
3	1	0.04	0
4	1	0.05	0
...	...	...	...
90995	364	2.46	0
90996	364	2.47	0
90997	364	2.48	0
90998	364	2.49	0
90999	364	2.50	0

91000 rows × 3 columns

```
import seaborn as sns
```

```
sns.lineplot(x=wheel_data.time, y=wheel_data.wheel)
```

```
<Axes: xlabel='time', ylabel='wheel'>
```

Day 2

```
1 dataset
```

```
xarray.Dataset
▶ Dimensions:      (trial: 364, time: 250, cell: 734, waveform_component: 3, sample: 82, probe: 384,
                   brain_area_lfp: 7, spike_id: 2446173)
▼ Coordinates:
    trial          (trial)                 int32   1 2 3 4 5 6 ... 360 361 362 363 ...
    time           (time)                  float64 0.01 0.02 0.03 ... 2.48 2.49 2.5
```

```
x = dataset[['pupil_x','pupil_y','pupil_area']].to_dataframe()
r = dataset['reaction_time'].to_dataframe()
```

```
    probe          (probe)                 int32   1 2 3 4 5 6 ... 380 381 382 383 ...
```

```
from matplotlib import pyplot as plt
```

```
    spike_id       (spike_id)              int32
```

```
!pip install --upgrade plotly
```

```
    contrast_right (trial)                 int8    0 50 50 0 100 0 ... 25 25 50 0 1
```

```
!pip install jupyter-dash
```

```
    stim_onset     (trial)                 float64 0.5 0.5 0.5 0.5 ... 0.5 0.5 0.5 0.5
```

```
x
```

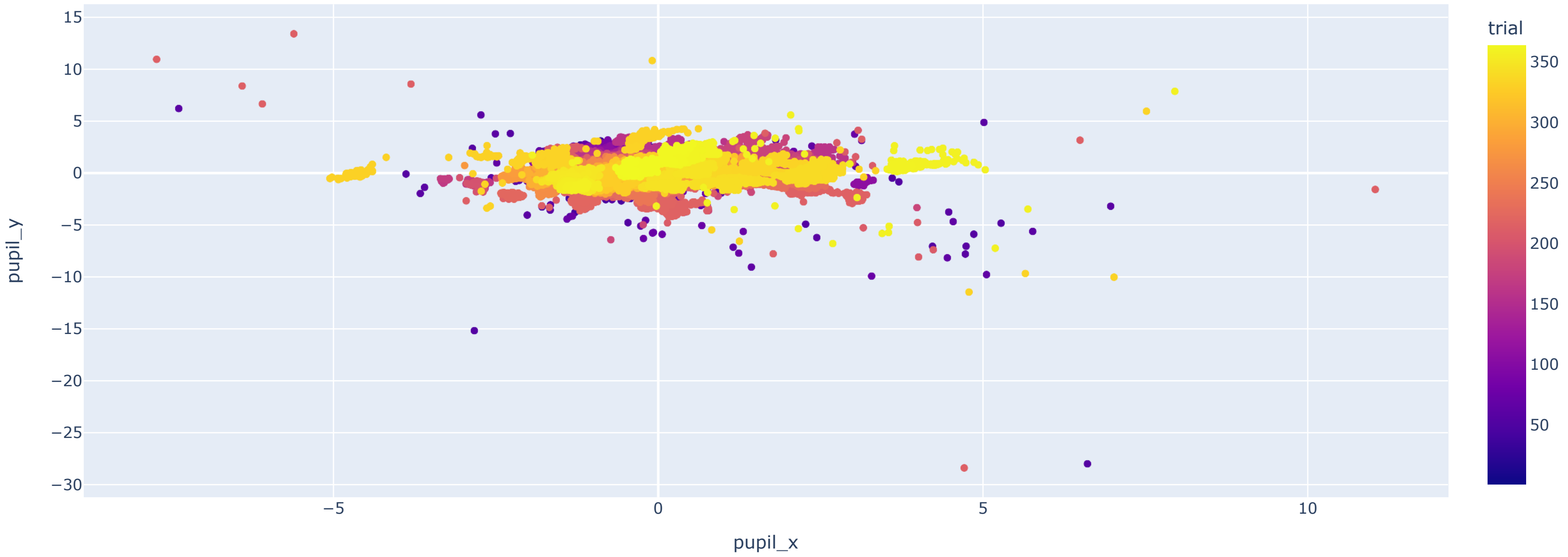
trial	time	pupil_x	pupil_y	pupil_area
1	0.01	0.812856	0.664247	0.165841
	0.02	0.778162	0.612752	0.158668
	0.03	0.812352	0.632769	0.169667
	0.04	0.760796	0.779100	0.172513
	0.05	0.790401	0.680980	0.176774
...	...	...	...	...
364	2.46	-0.446190	0.230169	0.117658
	2.47	-0.472352	0.359197	0.120560
	2.48	-0.453607	0.130451	0.113835
	2.49	-0.486026	0.239184	0.109935
	2.50	-0.491644	0.270002	0.119929

91000 rows × 3 columns

```
    lfp            (brain_area_lfp, trial, time)  float64 -2.851 -4.04 -4.195 ... 16.04 5....
```

```
import plotly.express as px

fig = px.scatter_3d(x, x='pupil_x', y='pupil_y', z='pupil_area')
fig.show()
```

```
fig = px.scatter(x1, x='pupil_x', y='pupil_y',color='trial')
fig.show()
```

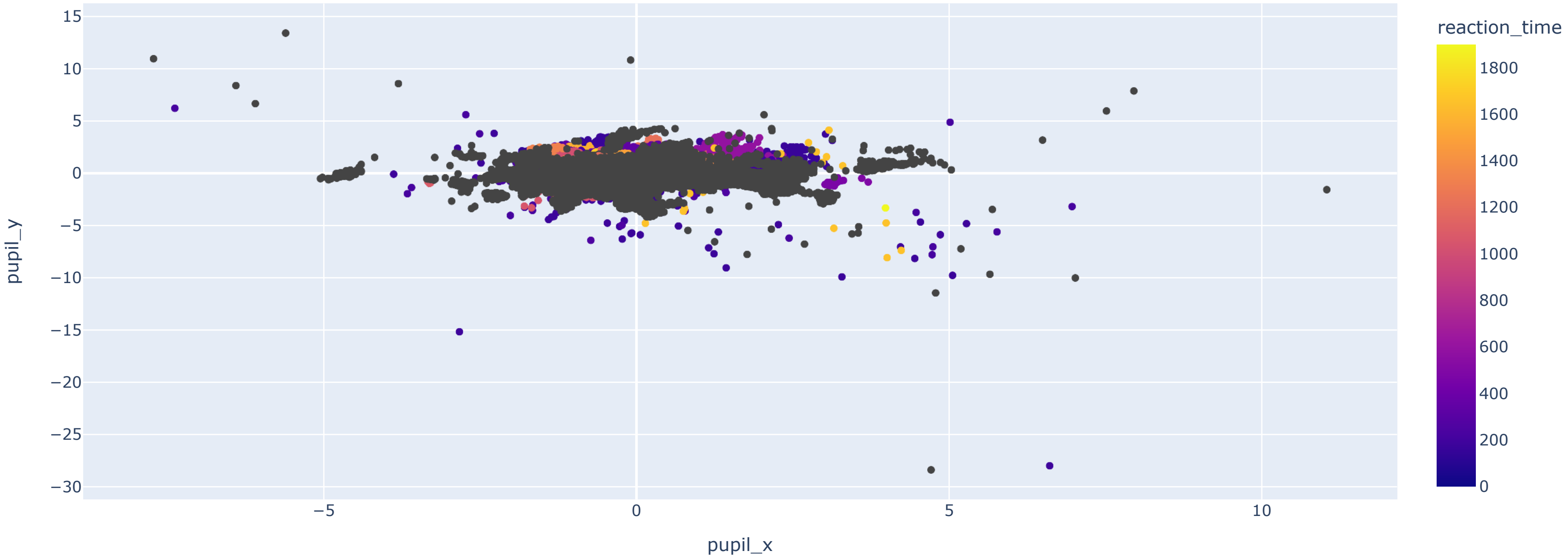


```
import pandas as pd
rt = pd.merge(x,r, on='trial')
rt
```

	pupil_x	pupil_y	pupil_area	reaction_time
trial				
1	0.812856	0.664247	0.165841	170.0
1	0.778162	0.612752	0.158668	170.0
1	0.812352	0.632769	0.169667	170.0
1	0.760796	0.779100	0.172513	170.0
1	0.790401	0.680980	0.176774	170.0
...	...	...	...	...
364	-0.446190	0.230169	0.117658	NaN
364	-0.472352	0.359197	0.120560	NaN
364	-0.453607	0.130451	0.113835	NaN
364	-0.486026	0.239184	0.109935	NaN
364	-0.491644	0.270002	0.119929	NaN

91000 rows × 4 columns

```
fig = px.scatter(rt, x='pupil_x', y='pupil_y',color='reaction_time')
fig.show()
```



1 dataset

xarray.Dataset

▶ Dimensions: (**trial**: 364, **time**: 250, **cell**: 734, **waveform_component**: 3, sample: 82, **probe**: 384, **brain_area_lfp**: 7, **spike_id**: 2446173)

▼ Coordinates:

trial	(trial)	int32	1 2 3 4 5 6 ... 360 361 362 363 ...	
time	(time)	float64	0.01 0.02 0.03 ... 2.48 2.49 2.5	

```
spk = dataset[['spike_trial','spike_time','spike_cell']].to_dataframe()
spk
```

spike_id	spike_trial	spike_time	spike_cell
1	21	0.267635	1
2	21	2.308335	1
3	31	0.853476	1
4	37	2.197501	1
5	43	2.452692	1
...	...	...	...
2446169	364	0.266459	734
2446170	364	0.587060	734
2446171	364	0.618560	734
2446172	364	2.188634	734
2446173	364	2.399335	734

2446173 rows × 3 columns

spike_rate (cell, trial, time) int8 0 0 0 0 0 0 0 0 ... 0 0 0 0 0 0 0 0

```
V = dataset['brain_area'].to_dataframe()
cell_label = V[V['brain_area']=='VISp'].index
```

ccf_dv (cell) float64 2.445e+03 1.516e+03 ... 4.094...

```
all_cell = spk[spk['spike_cell'].isin(cell_label)]
```

brain_groups (cell) object 'non-visual cortex' ... 'hippoca

spike_id	spike_trial	spike_time	spike_cell
841385	1	0.050555	364
841386	1	0.108489	364
841387	1	0.172022	364
841388	1	0.376223	364

```python
average_spike_time = all_cell.groupby(['spike_trial','spike_cell'])[['spike_time']].mean()
average_by_cell = average_spike_time.groupby('spike_trial')['spike_time'].mean()
average_by_cell
```

```
spike_trial
1      1.100715
2      1.362811
3      1.211432
4      1.339632
5      1.139493
         ...
360    1.218692
361    1.161034
362    1.214097
363    1.199479
364    1.118665
Name: spike_time, Length: 364, dtype: float32
```

```python
ft = dataset['feedback_type'].to_dataframe()
```

```python
p_ft = ft[ft['feedback_type'] == 1].index.to_list()
f1 = average_by_cell[p_ft]
```

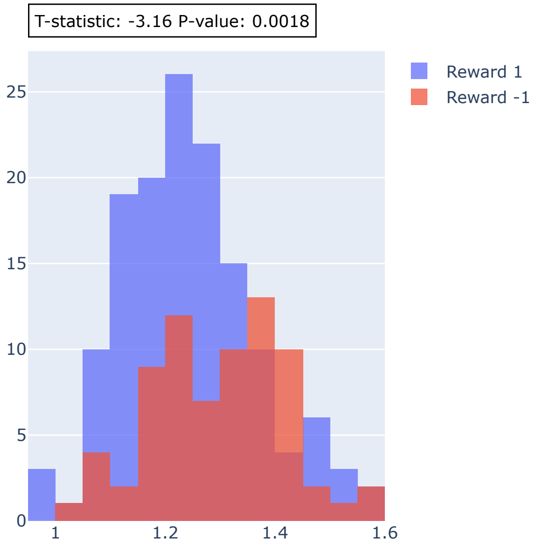
```python
p_ft = ft[ft['feedback_type'] == -1].index.to_list()
```

```python
f2 = average_by_cell[p_ft]
```

```python
from scipy.stats import ttest_ind

t_stat, p_val = ttest_ind(f1, f2)

```

```python
import plotly.graph_objects as go

# Create histogram
fig = go.Figure()

# Add traces (data) for the histograms
fig.add_trace(go.Histogram(x=f1, name='Reward 1', opacity=0.75,xbins=dict(size=0.05)))
fig.add_trace(go.Histogram(x=f2, name='Reward -1', opacity=0.75,xbins=dict(size=0.05)))

# Overlay histograms
```

```
fig.update_layout(barmode='overlay')

fig.add_annotation(
    x=0, y=1.1, # You might want to adjust these coordinates for better placement
    xref="paper", yref="paper", # Use "paper" coordinates for relative positioning
    text=f"T-statistic: {t_stat:.2f}\nP-value: {p_val:.4f}",
    showarrow=False,
    font=dict(size=12, color="black"),
    bordercolor="black",
    borderwidth=1,
    borderpad=4,
    bgcolor="white"
)
# Display the figure
fig.show()

```



```
SUB = dataset['brain_area'].to_dataframe()
cell_label = SUB[SUB['brain_area']=='SUB'].index
```

```
rs_spk = dataset['spike_rate'].to_dataframe().reset_index()
```

```
sub_cell = rs_spk[rs_spk['cell'].isin(cell_label)]
```

```
average_spike_time = sub_cell.groupby(['trial','cell'])[['spike_rate']].mean()
average_spike_time
```

```
average_by_cell = average_spike_time.groupby('trial')['spike_rate'].mean()
average_by_cell
```

```
trial
1      0.079307
2      0.067573
3      0.055947
4      0.056853
5      0.063520
         ...
360    0.090987
361    0.073333
362    0.083840
363    0.090827
364    0.085600
Name: spike_rate, Length: 364, dtype: float64
```

```
rt_df = dataset['reaction_time'].to_dataframe()
```

```
st_df = dataset[['spike_time','spike_cell','spike_trial']].to_dataframe()

```

```
st_cell_df = st_df[st_df['spike_cell'].isin(cell_label)]
average_spike_time = st_cell_df.groupby(['spike_trial','spike_cell'])[['spike_time']].mean()
average_spike_time = average_spike_time.groupby('spike_trial')['spike_time'].mean()
average_spike_time
```

```
spike_trial
1      1.134605
2      1.268446
3      1.088225
4      1.205556
5      1.177878
         ...
360    1.251748
361    1.258168
362    1.285870
363    1.120888
364    1.388254
Name: spike_time, Length: 364, dtype: float32
```

```
import pandas as pd
import numpy as np
```

```
df = pd.DataFrame({
    'reaction_time': rt_df.values.flatten(),
    'spike_time': average_spike_time,
    'spike_rate': average_by_cell,
})
```

```
df = df.reset_index()
df
```

	level_0	index	reaction_time	spike_time	spike_rate
0	0	1	170.0	1.134605	0.079307
1	1	2	230.0	1.268446	0.067573
2	2	3	200.0	1.088225	0.055947
3	3	4	860.0	1.205556	0.056853
4	4	5	140.0	1.177878	0.063520
...	...	...	...	...	...
359	359	360	NaN	1.251748	0.090987
360	360	361	NaN	1.258168	0.073333
361	361	362	NaN	1.285870	0.083840
362	362	363	NaN	1.120888	0.090827

364 rows × 5 columns

```
ft = dataset['feedback_type'].to_dataframe()
```

```
p_ft = ft[ft['feedback_type'] == 1].index.to_list()

f1 = df.iloc[p_ft,:]
f1
```

	level_0	index	reaction_time	spike_time	spike_rate
1	1	2	230.0	1.268446	0.067573
2	2	3	200.0	1.088225	0.055947
3	3	4	860.0	1.205556	0.056853
4	4	5	140.0	1.177878	0.063520
9	9	10	200.0	1.082780	0.070240
...	...	...	...	...	...
191	191	192	inf	1.193350	0.073280
194	194	195	inf	1.251339	0.060587
198	198	199	inf	1.114188	0.064107
200	200	201	1370.0	1.427714	0.066933
201	201	202	inf	1.275791	0.066133

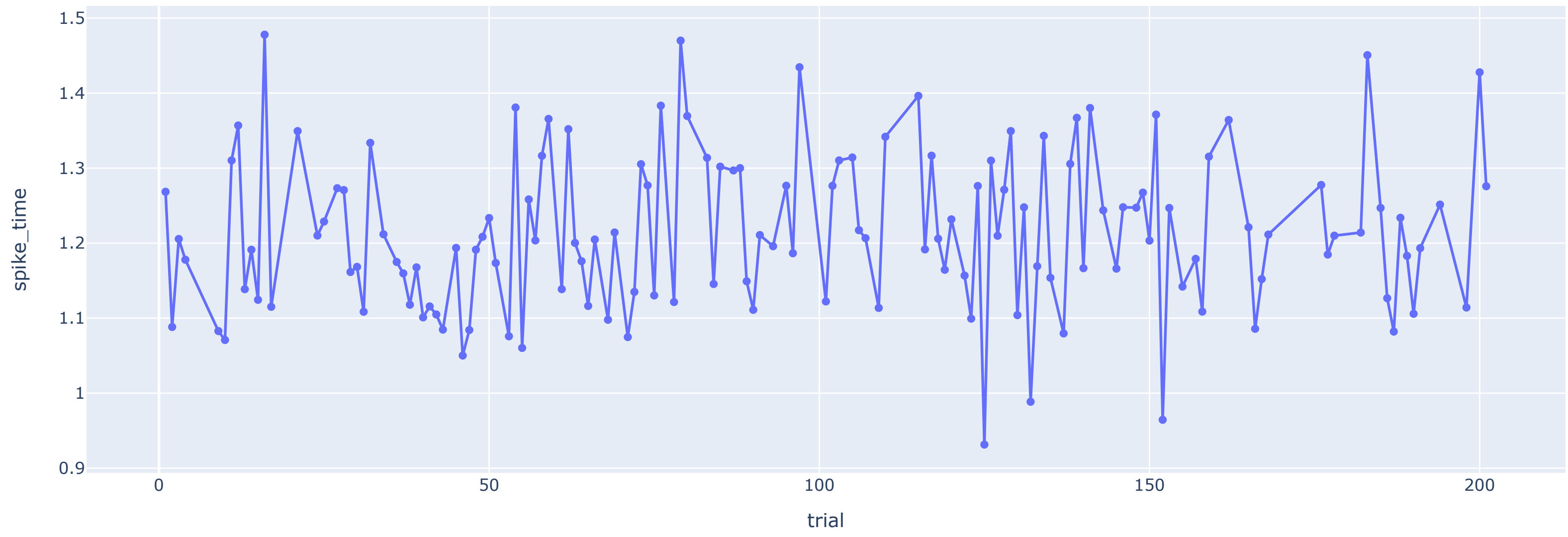
141 rows × 5 columns

```
p_ft = ft[ft['feedback_type'] == -1].index.to_list()

f2 = df.iloc[p_ft,:]
f2
```

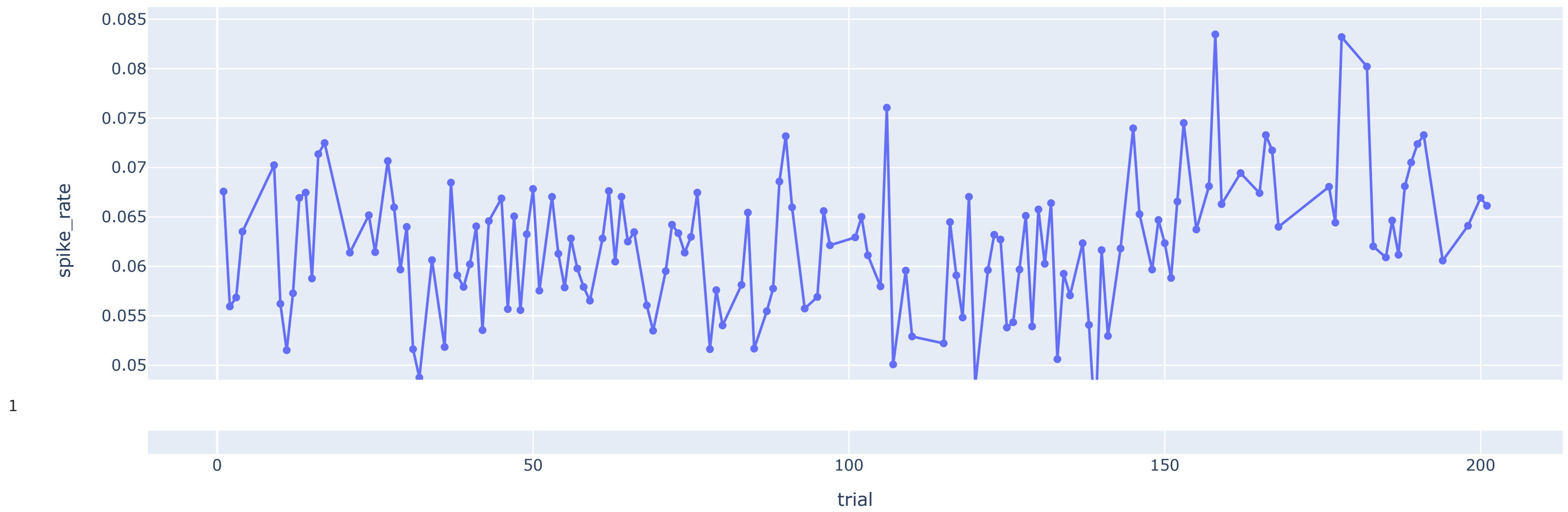
	level_0	index	reaction_time	spike_time	spike_rate
5	5	6	1340.0	1.265247	0.058560
6	6	7	740.0	1.170172	0.062667
7	7	8	990.0	1.086859	0.053440
8	8	9	1180.0	1.454938	0.045440
18	18	19	920.0	1.245060	0.058613
...	...	...	...	...	...
210	210	211	1050.0	1.317677	0.070080
211	211	212	inf	1.281139	0.067413
212	212	213	inf	1.327681	0.072853
213	213	214	1350.0	1.294707	0.073440

```
# Create line plot
fig = go.Figure(data=go.Scatter(x=f1.index, y=f1.reaction_time, mode='lines+markers'))

# Add title and labels
fig.update_layout(
                  xaxis_title='trial',
                  yaxis_title='reaction_time')

# Display the figure
fig.show()
```

```
# Create line plot
fig = go.Figure(data=go.Scatter(x=f1.index, y=f1.spike_time, mode='lines+markers'))

# Add title and labels
fig.update_layout( xaxis_title='trial',
                  yaxis_title='spike_time')

# Display the figure
fig.show()
```



```
# Create line plot
fig = go.Figure(data=go.Scatter(x=f1.index, y=f1.spike_rate, mode='lines+markers'))

# Add title and labels
fig.update_layout( xaxis_title='trial',
                  yaxis_title='spike_rate')

# Display the figure
fig.show()
```


0.085
0.08
0.075
0.07
0.065
0.06
0.055
0.05
spike_rate
1
0
50
100
150
200
trial