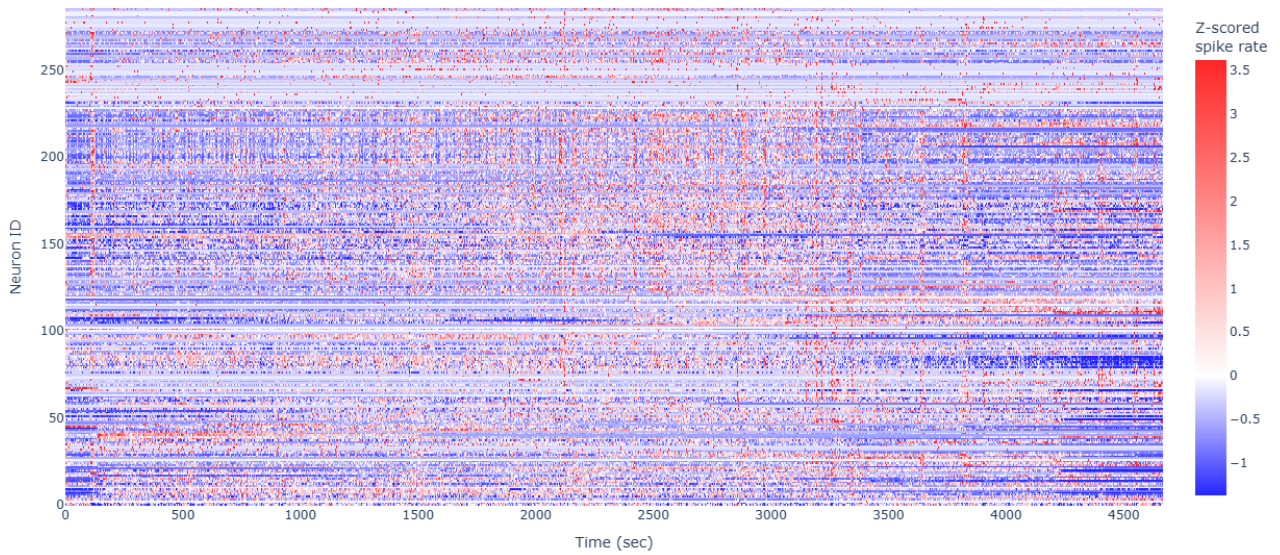
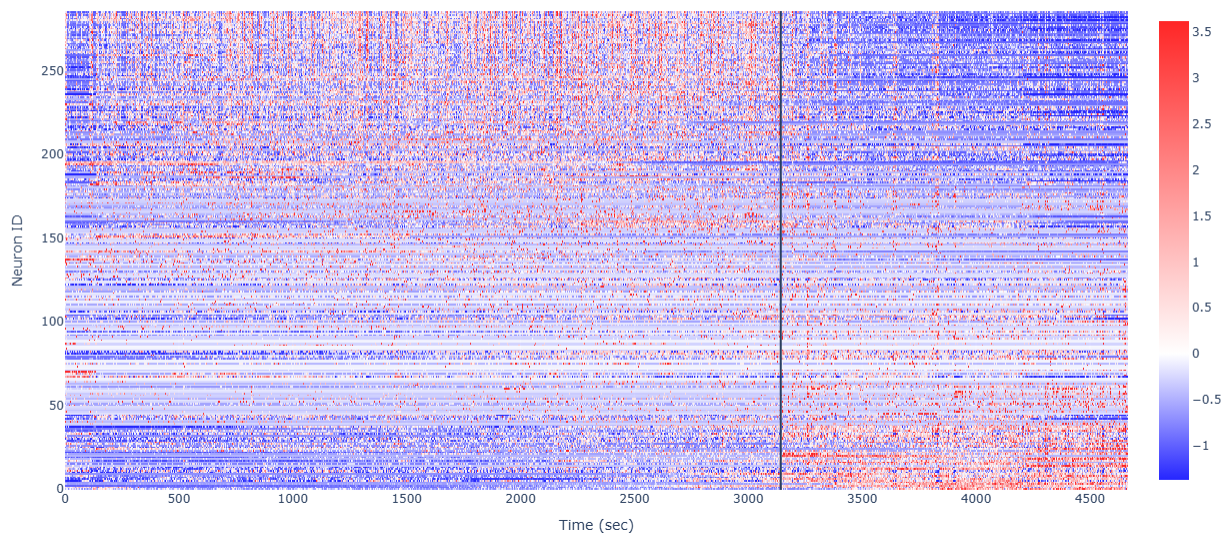


1.

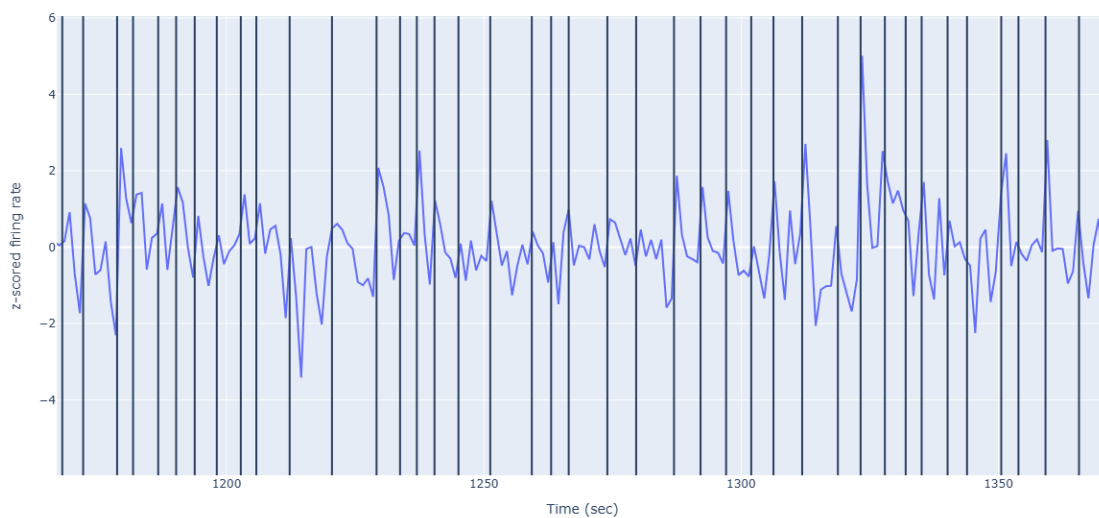


2. `sorted_neurons_indices = np.argsort(u[:, 0])`

`activity_arrayZ_sorted = activity_arrayTZ[sorted_neurons_indices, :]`



3. The first left singular vector projected onto the z-scored data matrix, i.e., $y = u[0, :] @ \text{activity_arrayTZ}$

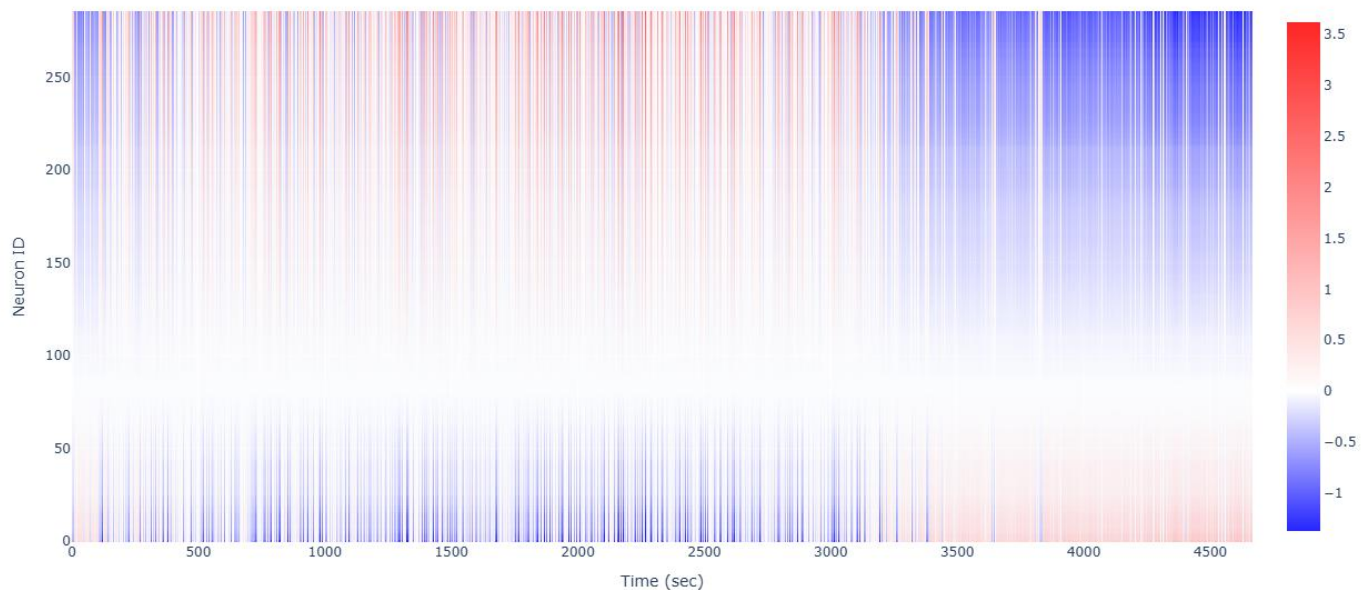


4.

```
def truncate_svd(u, s, vh, n_components):  
    truncated_svd = np.zeros((u.shape[0], vh.shape[1]), dtype=u.dtype)  
    for i in range(n_components):  
        truncated_svd += s[i]*np.outer(u[:, i], vh[i, :])  
    return truncated_svd
```

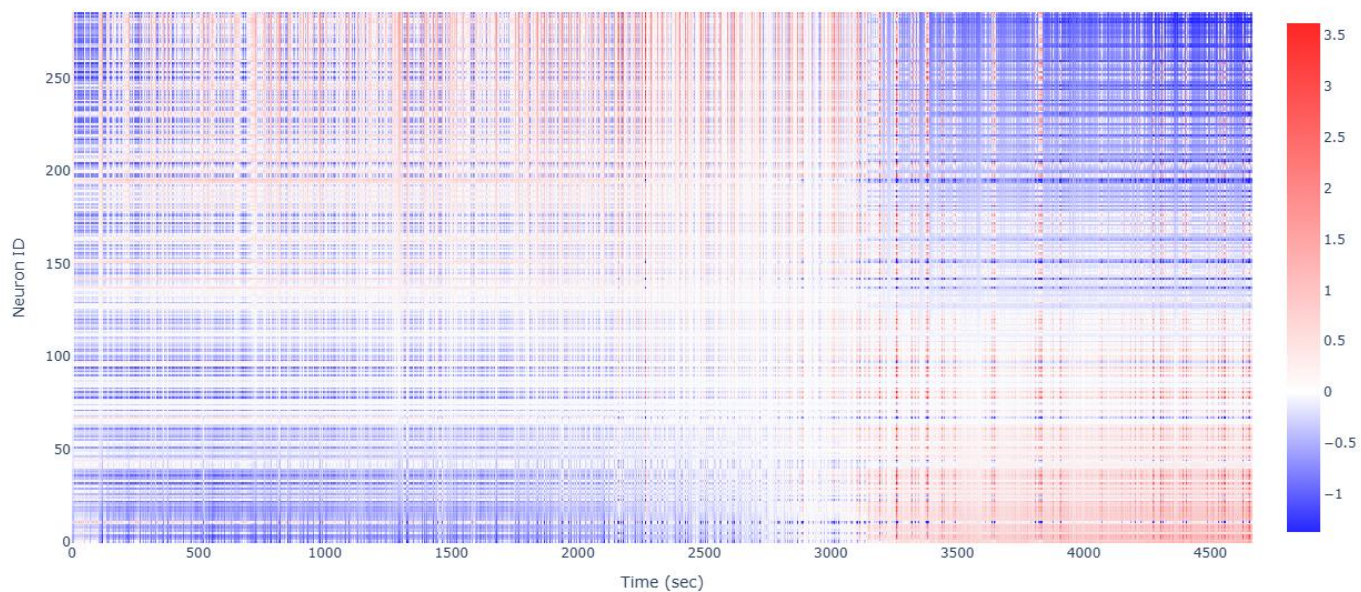
n_components=1

analytical error: 1084.57, empirical error: 1084.57



n_components=2

analytical error: 1023.43, empirical error: 1023.43



n_components=5

analytical error: 973.95, empirical error: 973.95

