

Embedding Analysis of Neuron Responses under Various Stimuli in dLGN and SC

Huimiao Chen
Nov 22, 2024

Cells in **dLGN** from **Multiple FOVs** with **36-Stim Set**
PCA & UMAP (reduce **neurons** dim)

Cells in **dLGN** from **Multiple FOVs** with **36-Stim Set**
PCA, UMAP, Manifold Embedding (reduce **stimuli** dim)

Cells in **dLGN** from **Multiple FOVs** with **48-Stim Set**
PCA & UMAP (reduce **neurons** dim)

Cells in **dLGN** from **Multiple FOVs** with **48-Stim Set**
PCA, UMAP, Manifold Embedding (reduce **stimuli** dim)

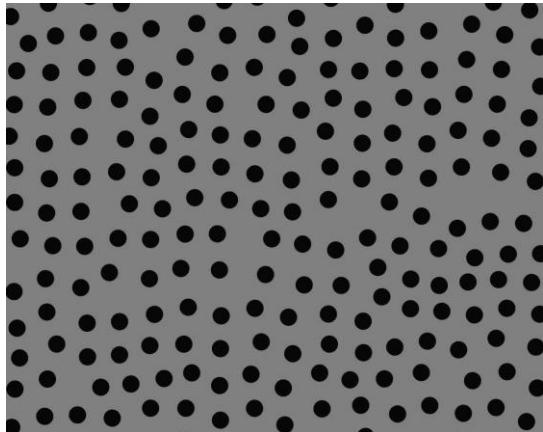
Cells in **SC** from **Multiple FOVs** with **48-Stim Set**
PCA & UMAP (reduce **neurons** dim)

Cells in **SC** from **Multiple FOVs** with **48-Stim Set**
PCA, UMAP, Manifold Embedding (reduce **stimuli** dim)

Cells in dLGN from Multiple FOVs with 36-Stim Set
PCA & UMAP (reduce neurons dim)

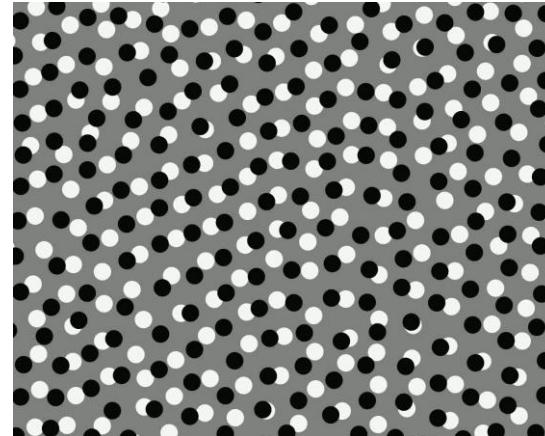
Stimuli

```
# dir: 0, 90, 180, 270
Gratings
neg_1flows
pos_1flows
neg_3flows
pos_3flows
pos_oppdir
neg_oppdir
posneg_samedir
posneg_oppdir
```

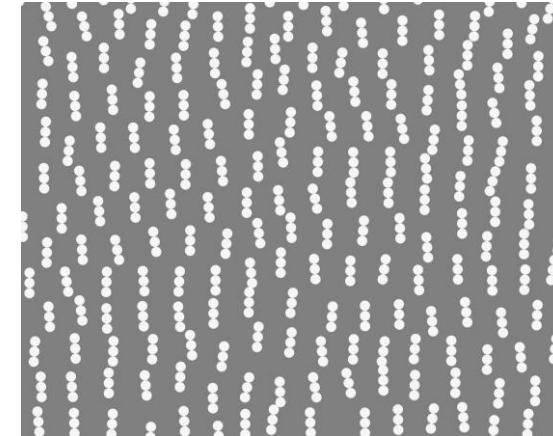


neg_1flows_180

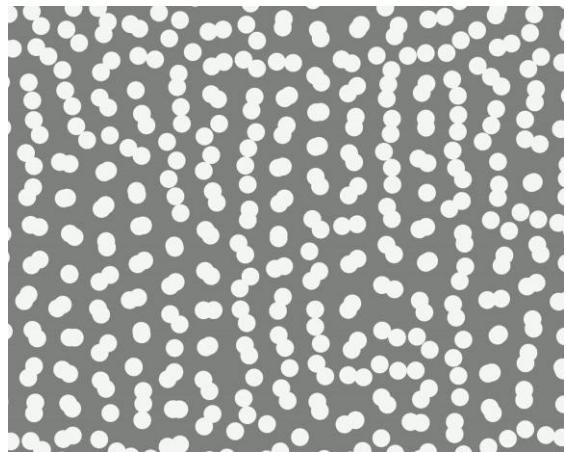
```
# 36 stimuli
label_names = [
    'neg_1flows_270', 'posneg_oppdir_270', 'pos_1flows_180', 'neg_3flows_180',
    'posneg_samedir_270', 'neg_oppdir_180', 'gratings_90', 'pos_oppdir_0',
    'pos_3flows_180', 'neg_1flows_0', 'pos_1flows_270', 'gratings_270',
    'posneg_oppdir_0', 'neg_3flows_90', 'pos_oppdir_90', 'posneg_samedir_180',
    'neg_oppdir_270', 'pos_3flows_270', 'neg_3flows_0', 'gratings_0',
    'posneg_samedir_90', 'pos_3flows_0', 'neg_1flows_90', 'posneg_oppdir_180',
    'neg_oppdir_0', 'pos_1flows_0', 'pos_oppdir_180', 'pos_oppdir_270',
    'pos_1flows_90', 'posneg_oppdir_90', 'posneg_samedir_0', 'gratings_180',
    'neg_1flows_180', 'neg_3flows_270', 'neg_oppdir_90', 'pos_3flows_90'
```



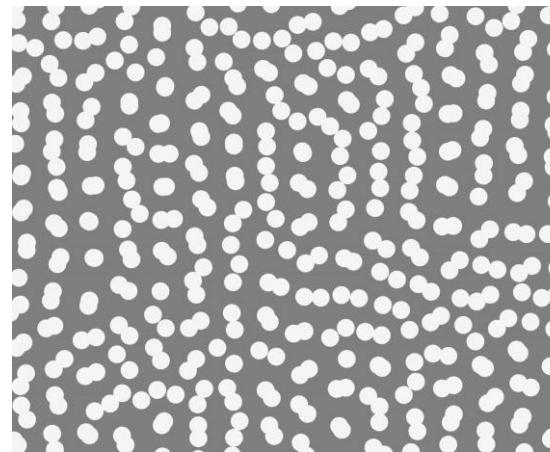
posneg_samedir_0



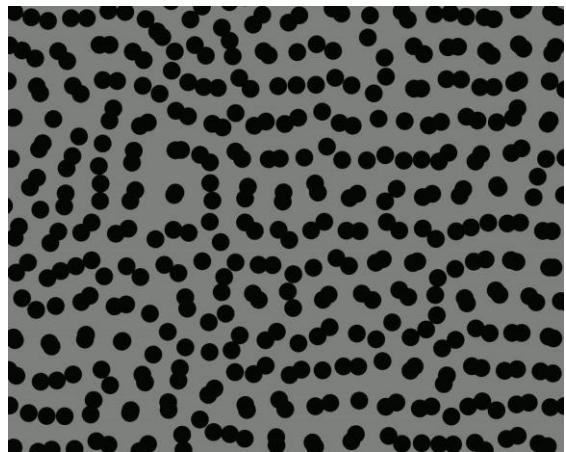
pos_3flows_0



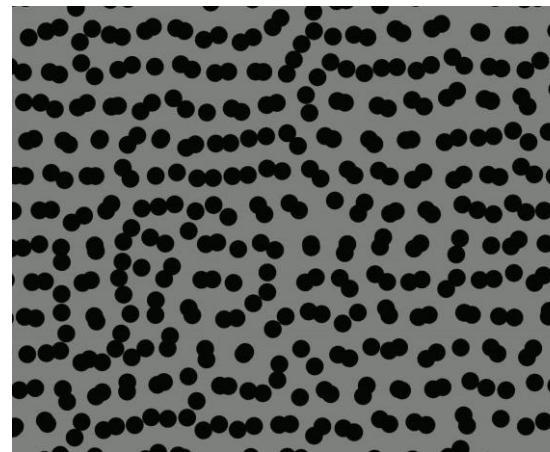
pos_oppdir_0



pos_oppdir_180



neg_oppdir_90



neg_oppdir_270

Preprocessed Data for one FOV:

2-D Matrix

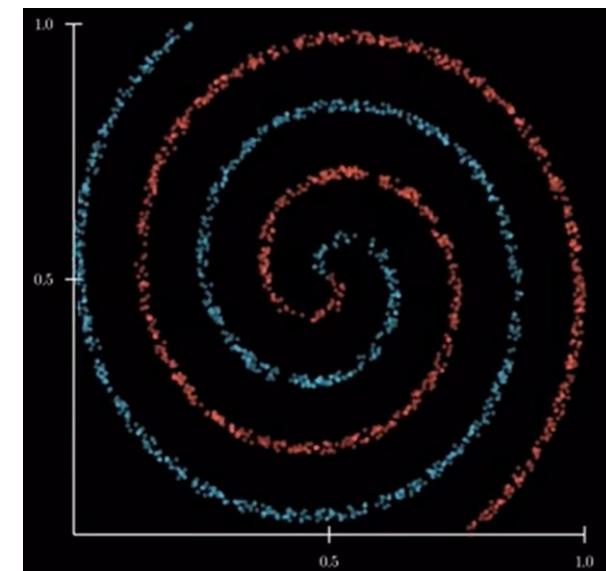
N_STIMS*N_SESSIONS*N_REPS by N_NEURONS

Multiple FOVs – stack together

```
exp_dLGN_36 = [  
    'CC182_240905_ThresholdedNaN3_Combined',  
    'CC182_240906_ThresholdedNaN3_Combined',  
    'CC182_240911_ThresholdedNaN3_Combined',  
    'CC182_240912_ThresholdedNaN3_Combined',  
    'CC182_240913_ThresholdedNaN3_Combined',  
    'CC185_240916_ThresholdedNaN3_Combined',  
    'CC185_240917_ThresholdedNaN3_Combined',  
    'CC185_240919_ThresholdedNaN3_Combined'  
]
```

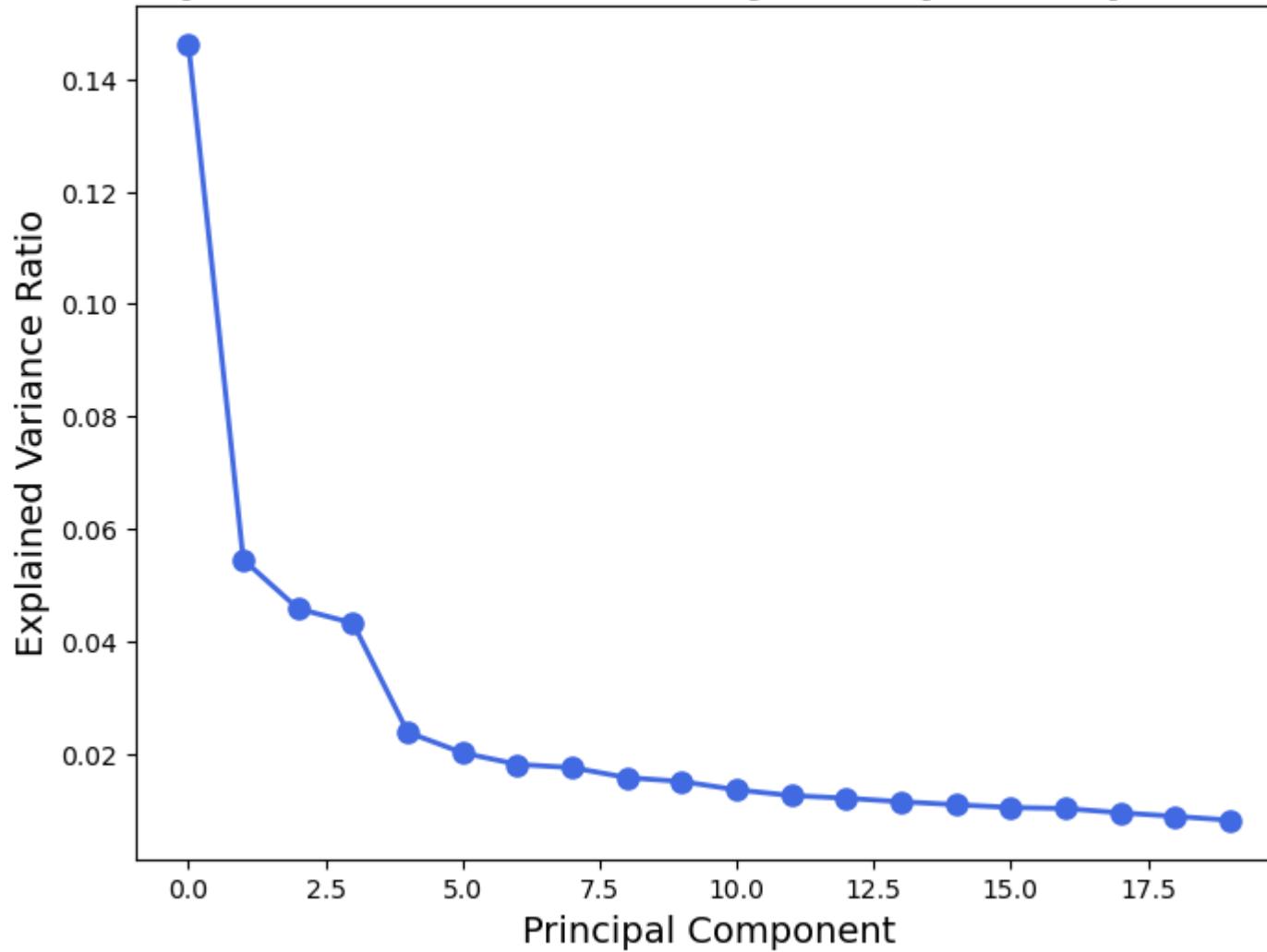
Why PCA & UMAP?

UMAP (Uniform Manifold Approximation and Projection) focuses on preserving the topological structure of the data. It constructs a graph of nearest neighbors in the original space and optimizing a low-dimensional embedding to maintain the graph's structure.

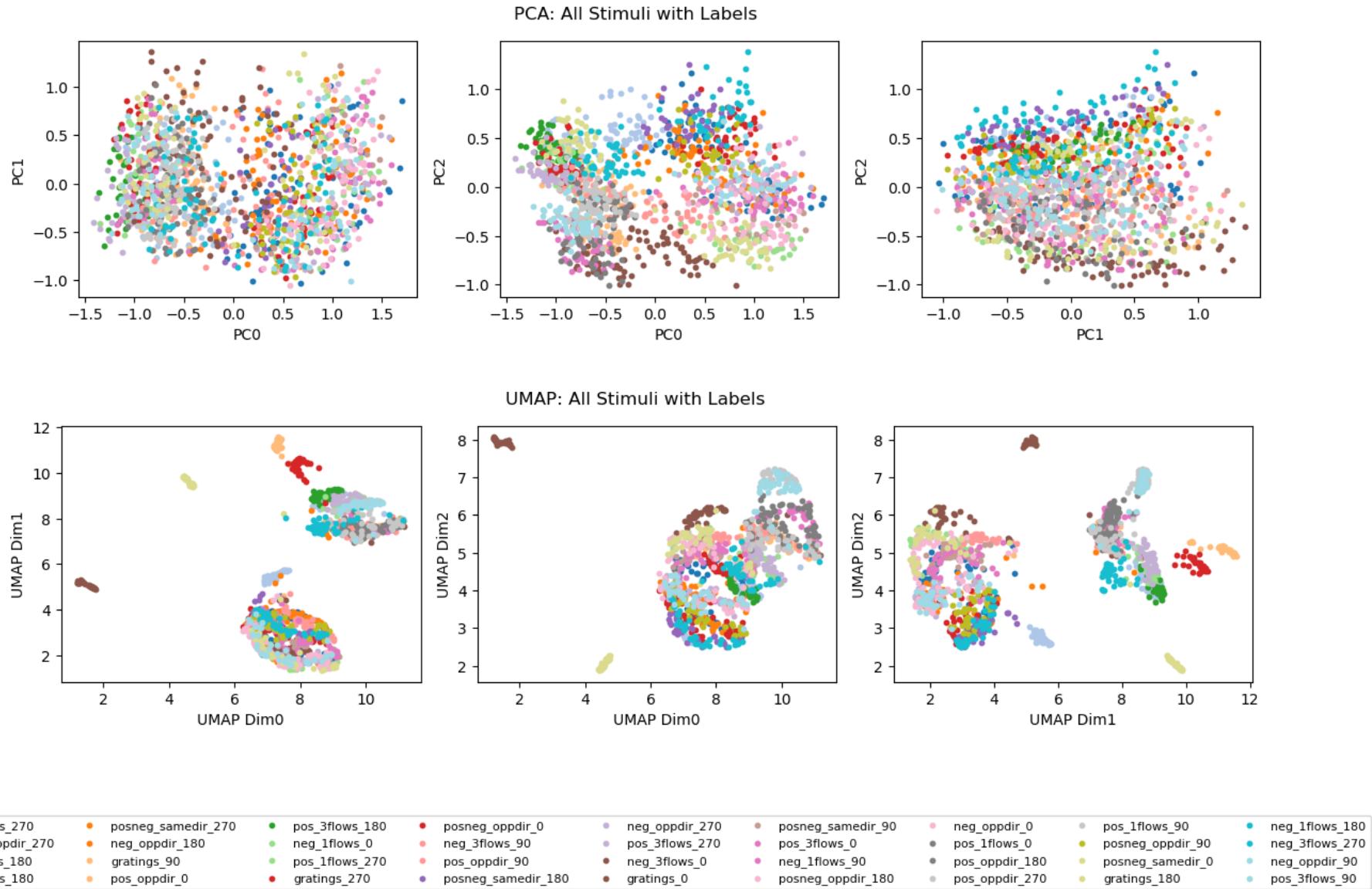


E.g., UMAP can capture the above structure (**NONLINEARITY**) in 1-D and PCA can't.

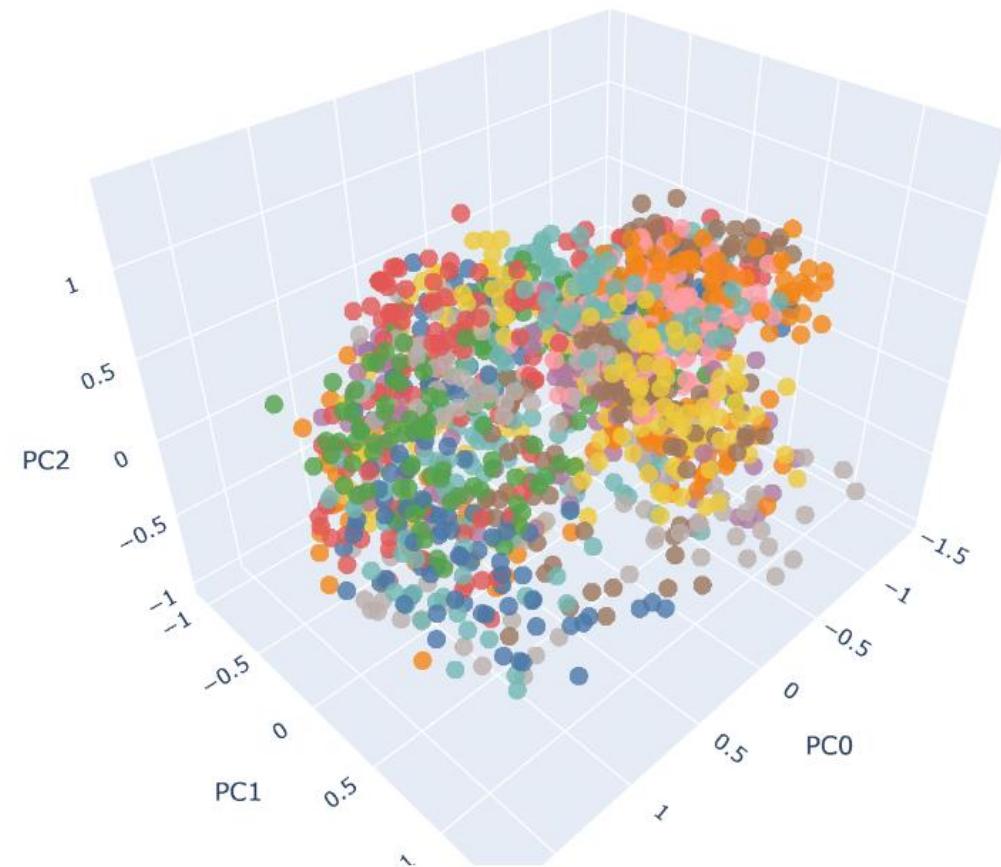
Explained Variance Ratio by Principal Components



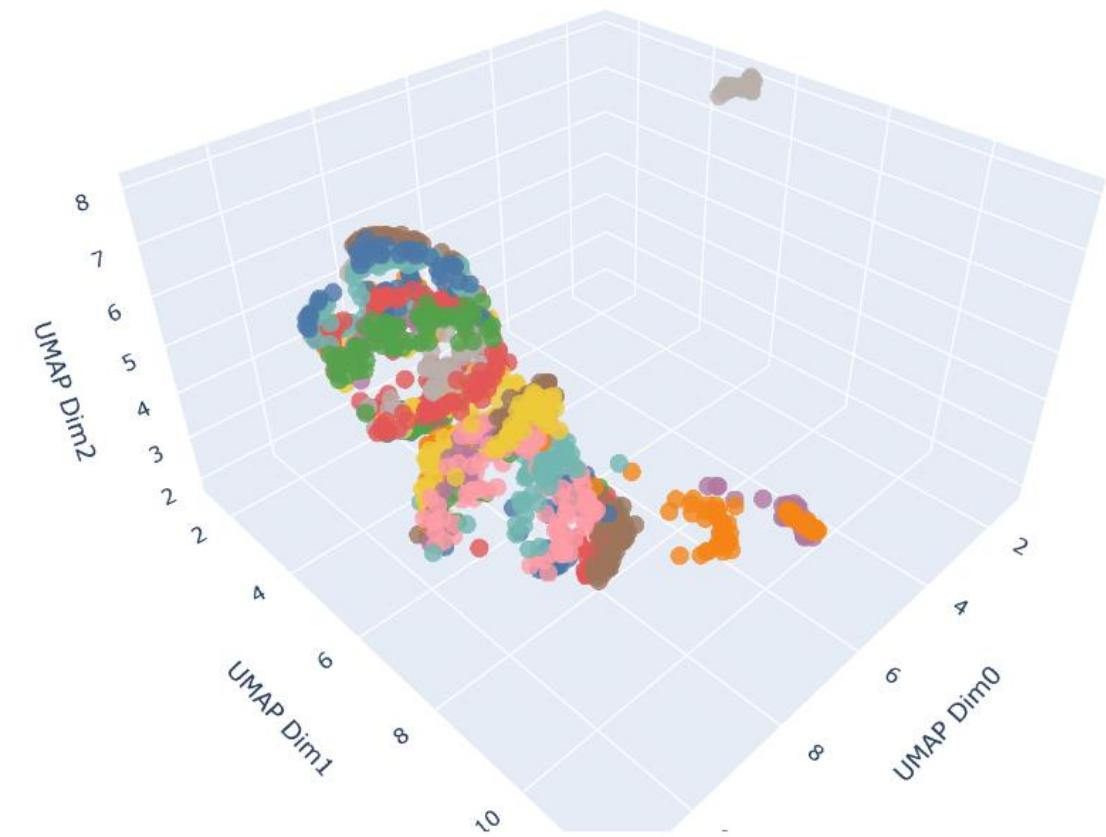
We use the first 3 PCs for plotting.



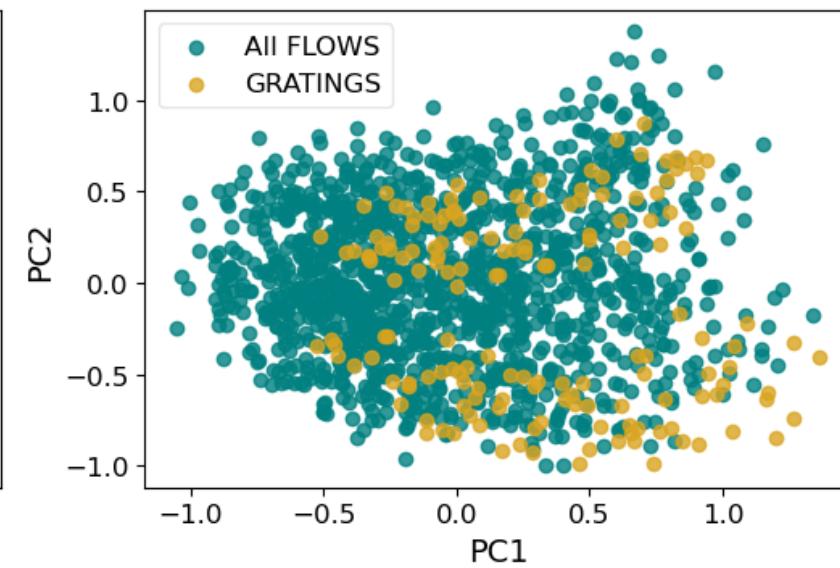
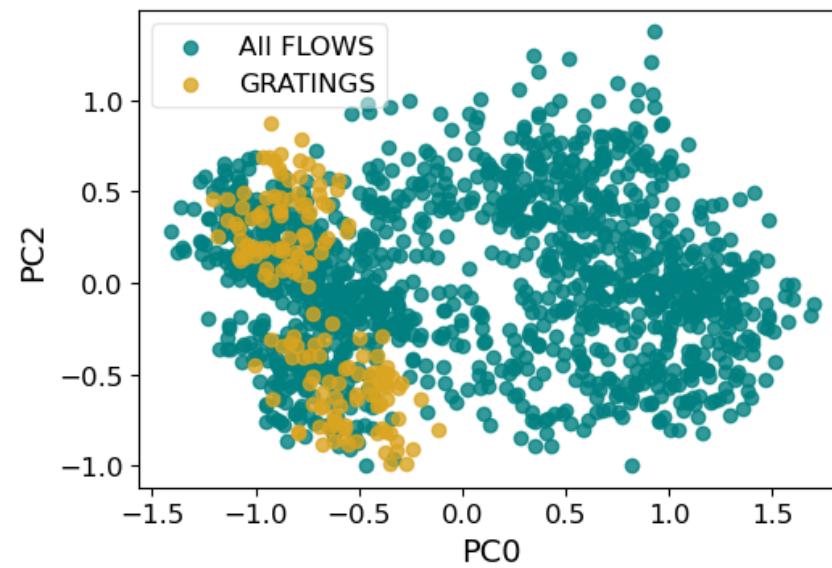
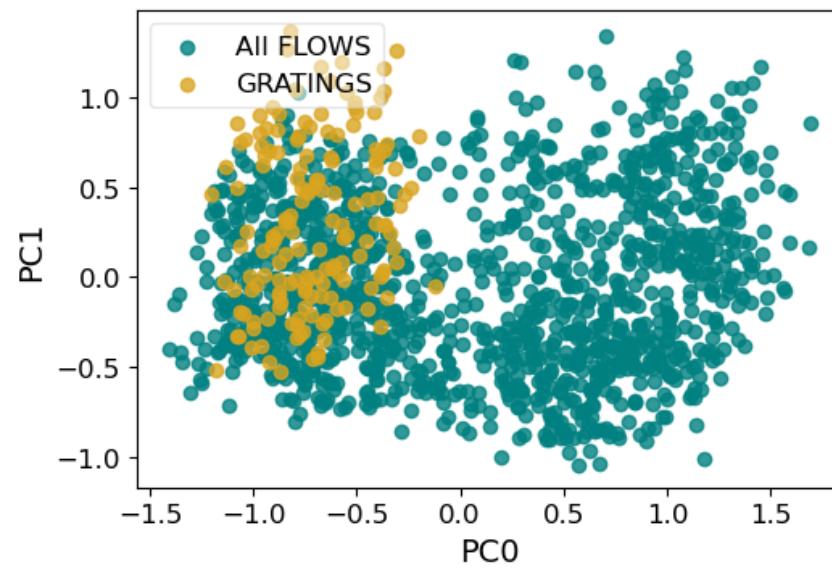
PCA: Interactive 3D Plot of All Stimuli



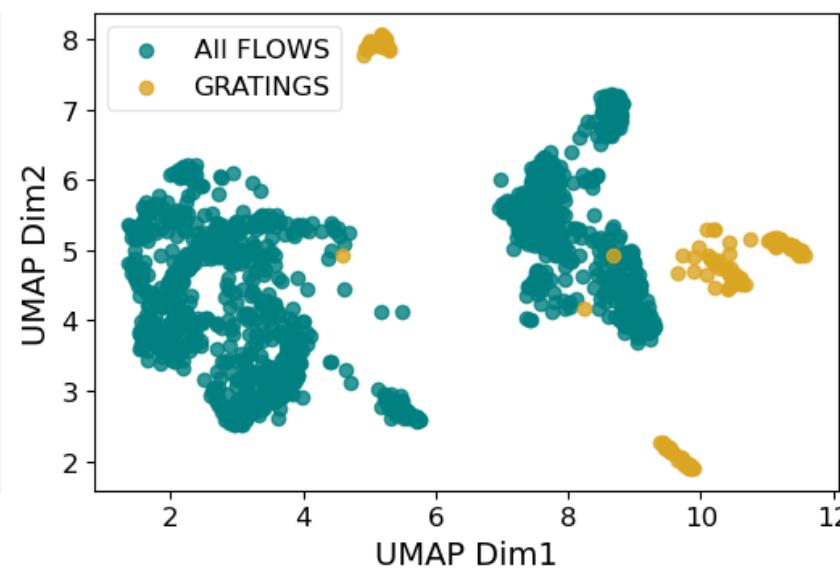
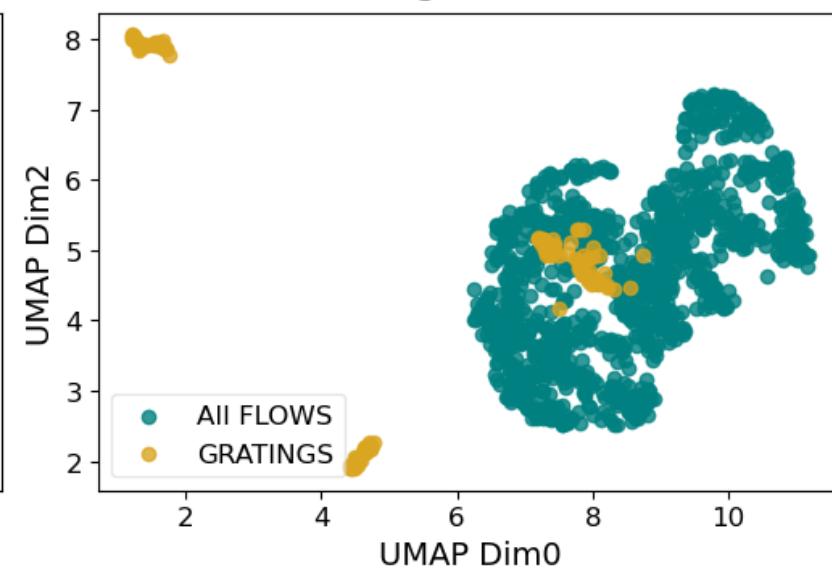
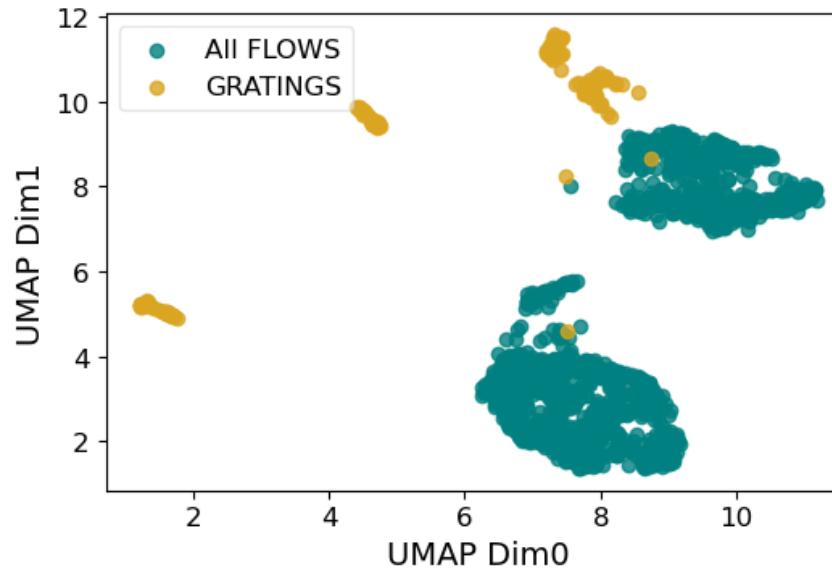
UMAP: Interactive 3D Plot of All Stimuli



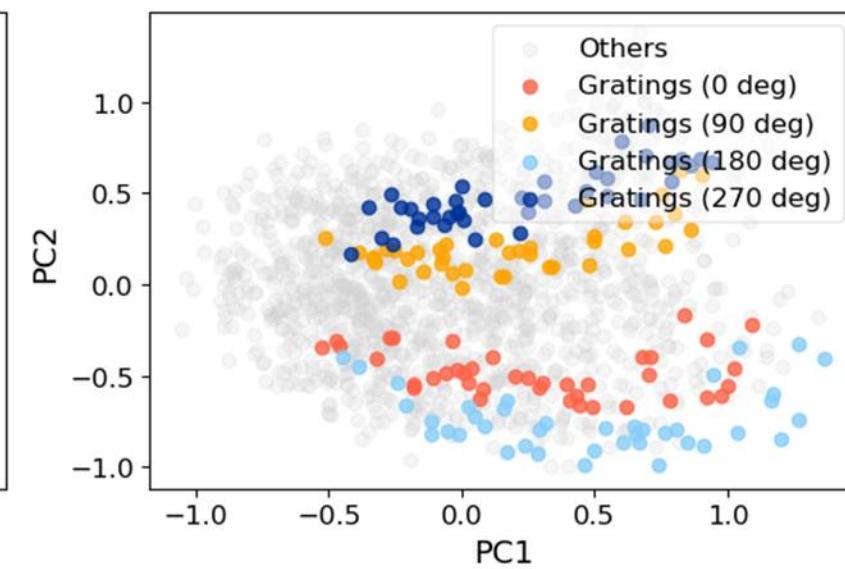
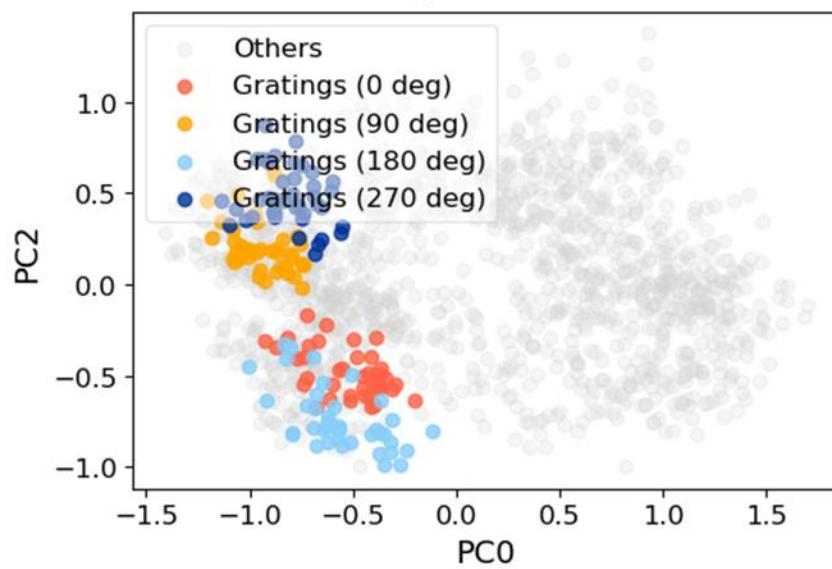
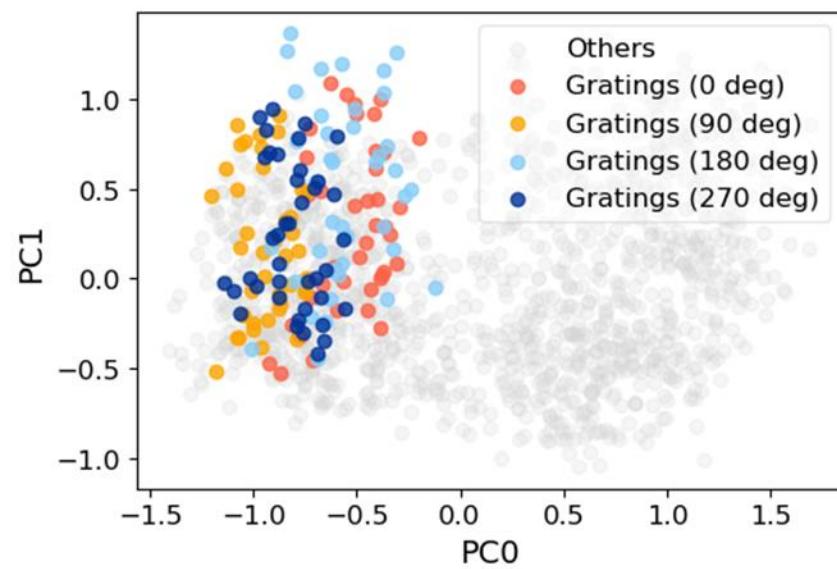
PCA: Gratings vs. All Flows



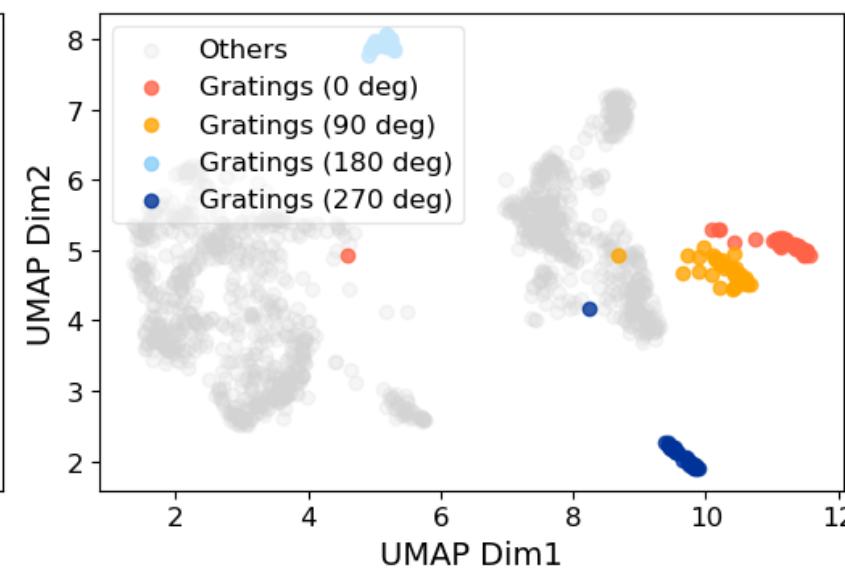
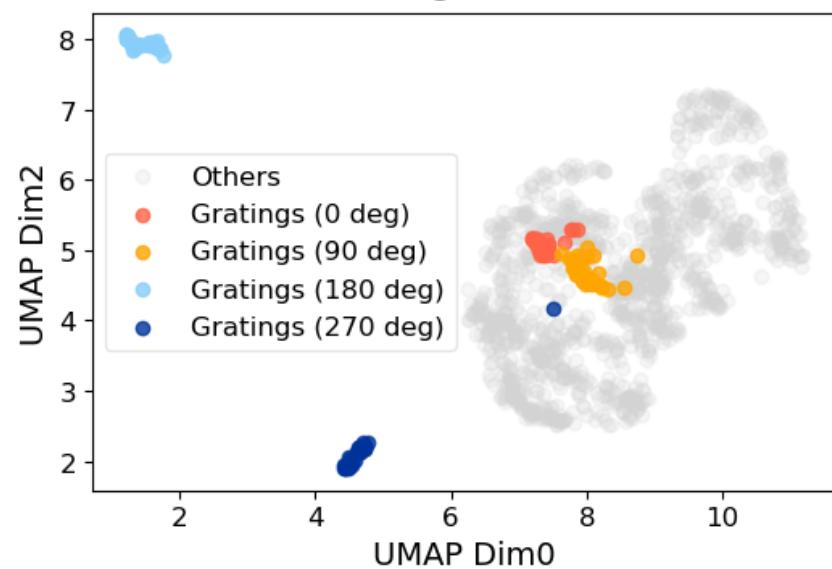
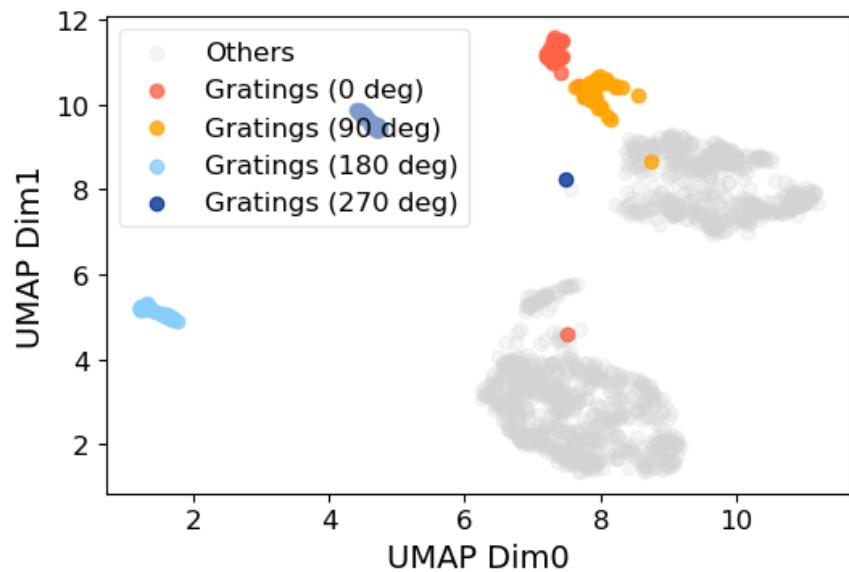
UMAP: Gratings vs. All Flows



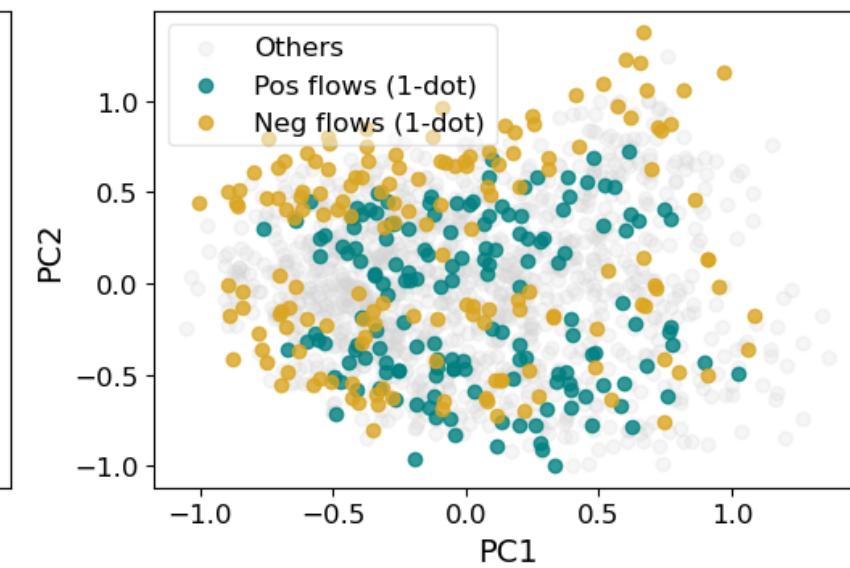
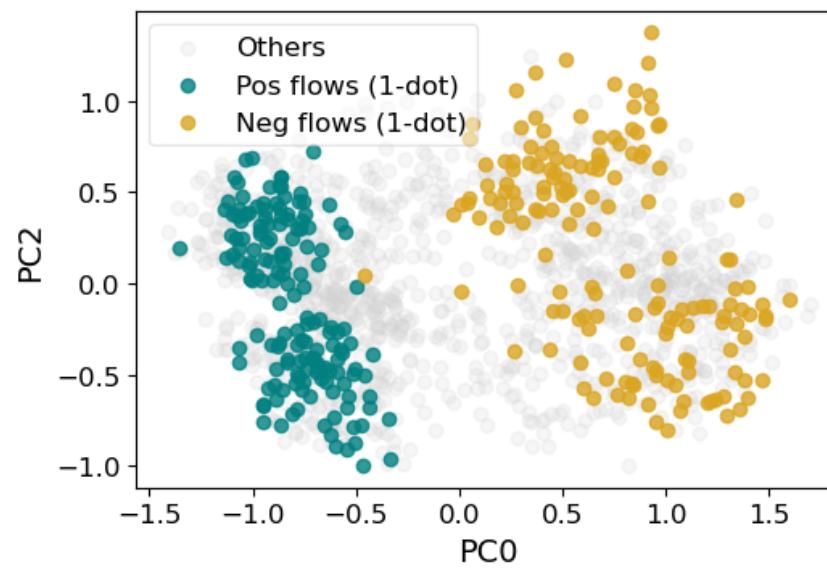
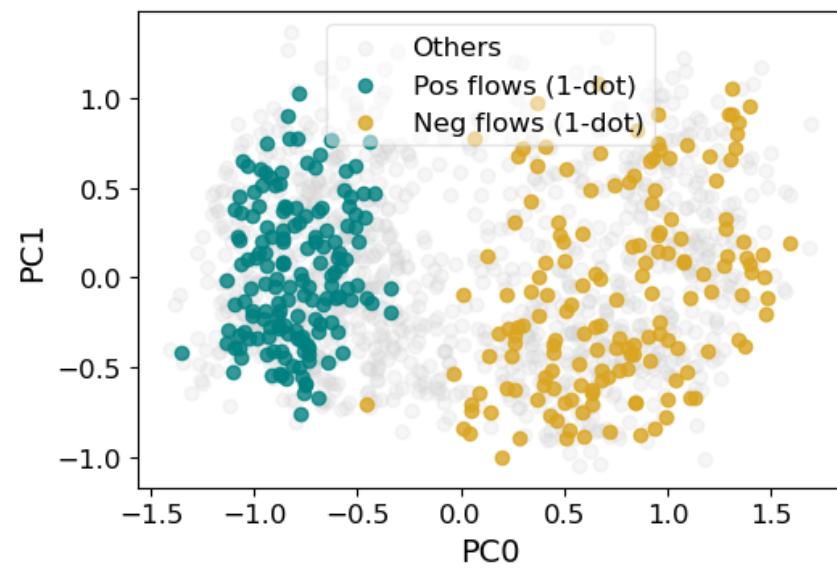
PCA: Gratings directions



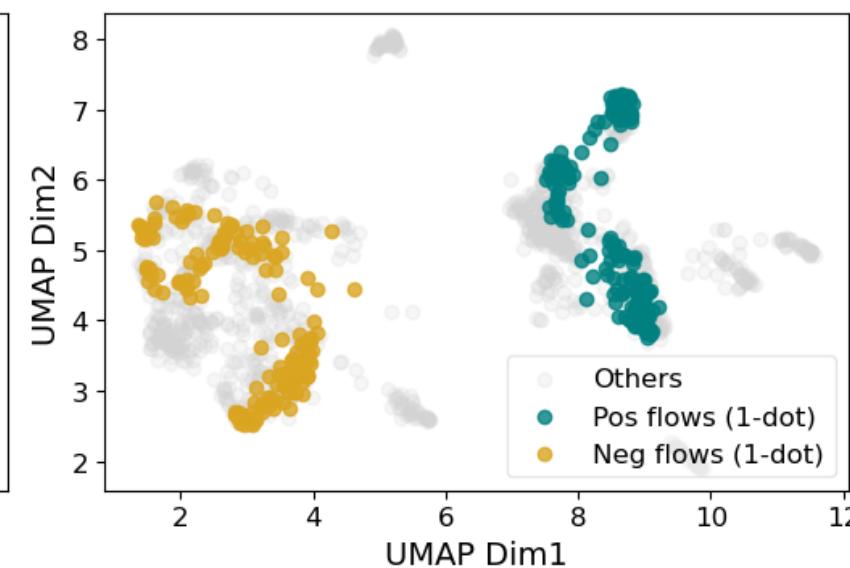
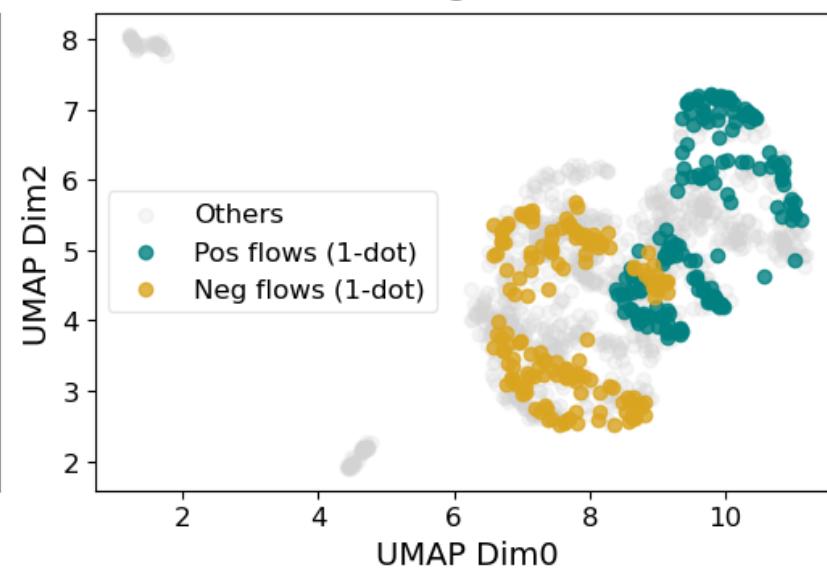
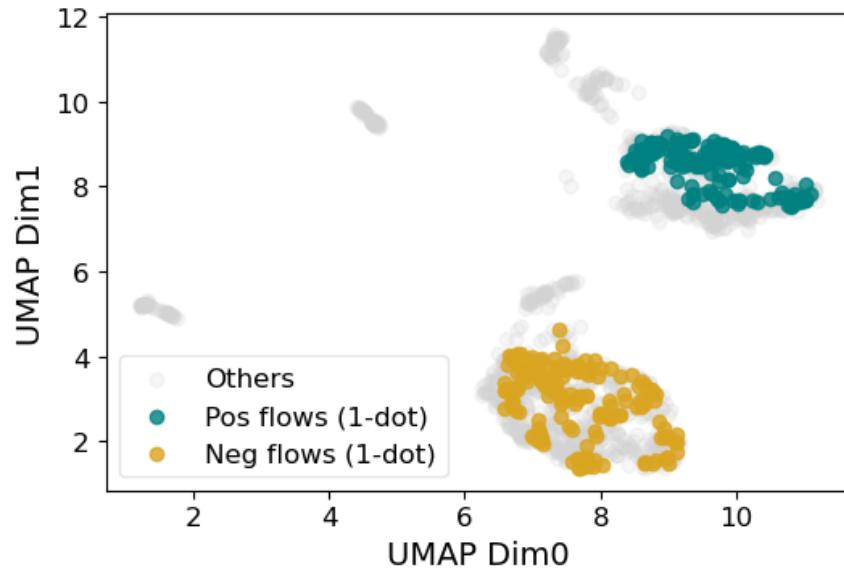
UMAP: Gratings directions



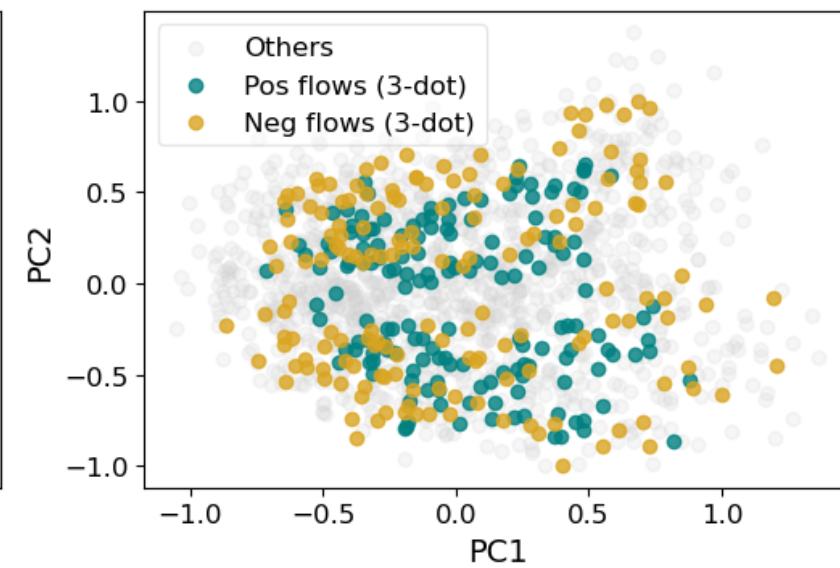
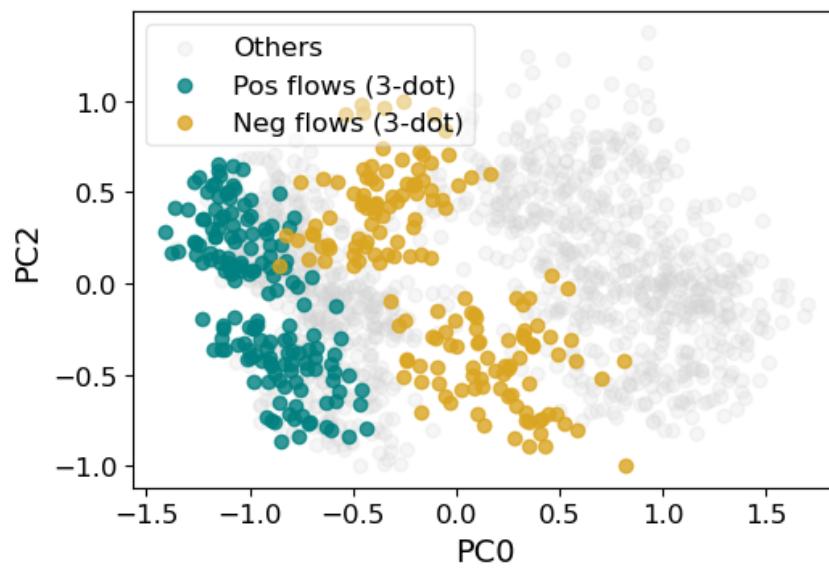
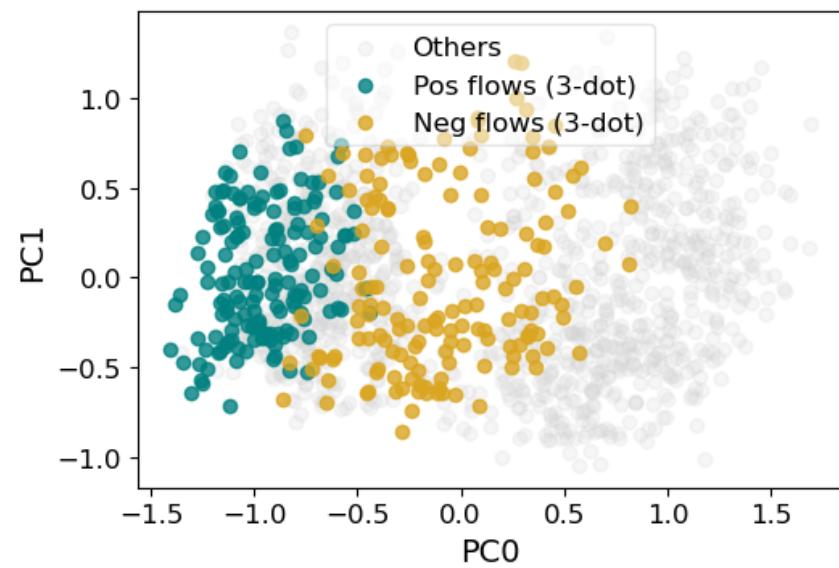
PCA: Pos vs neg flows (1-dot)



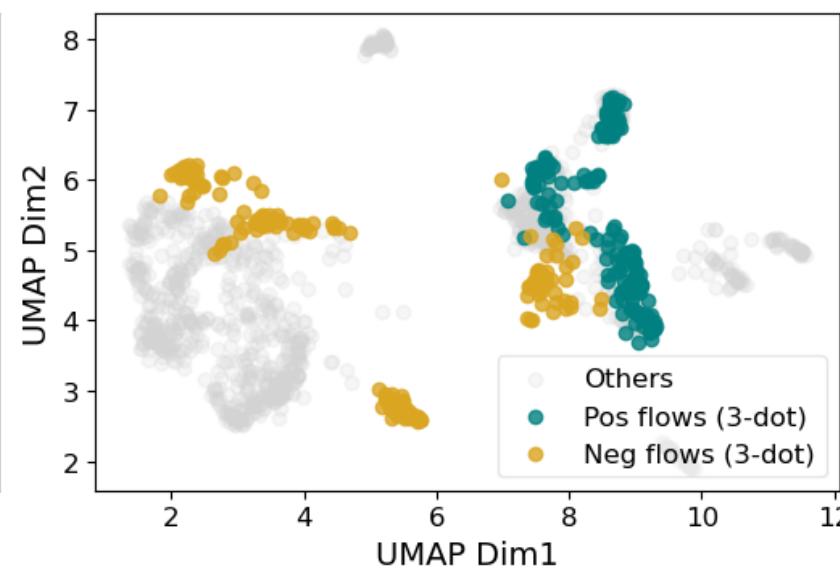
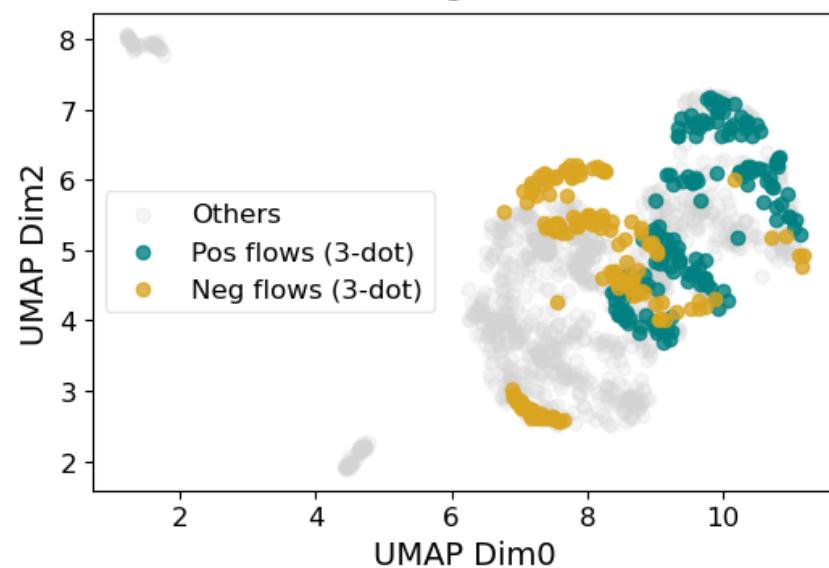
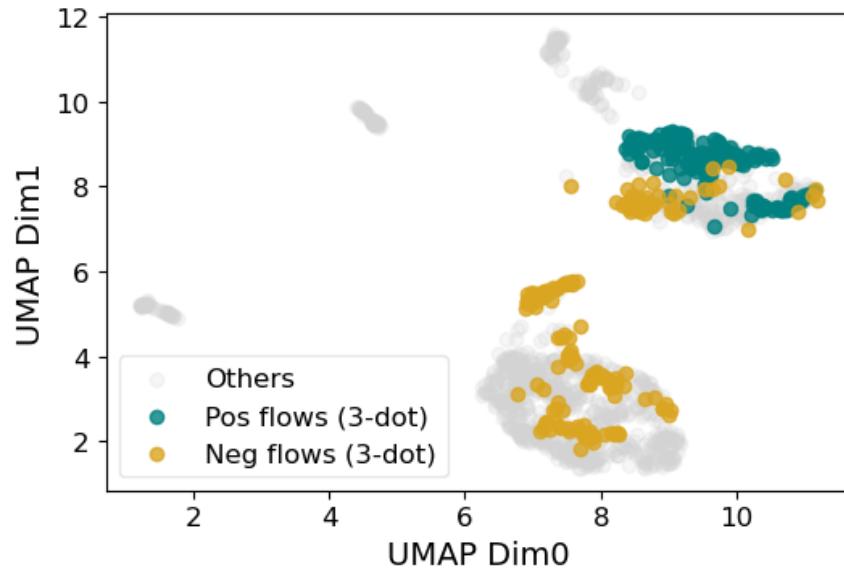
UMAP: Pos vs neg flows (1-dot)



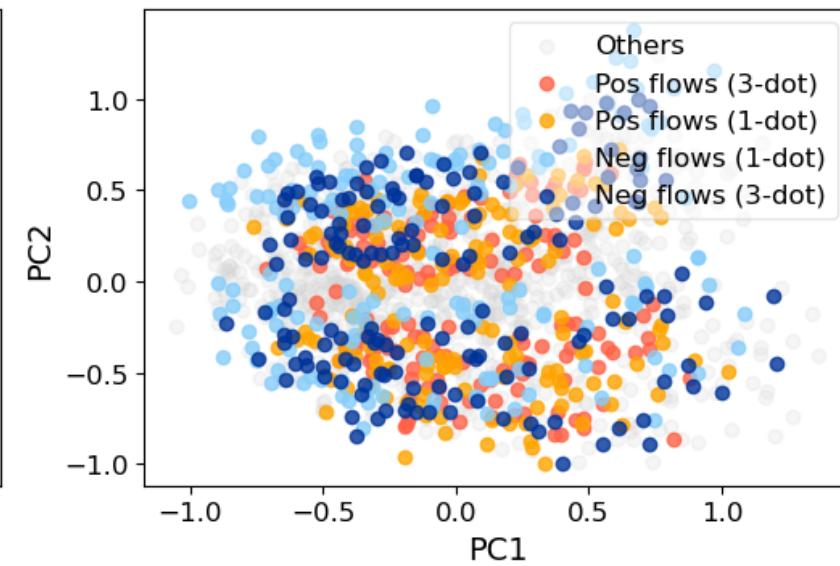
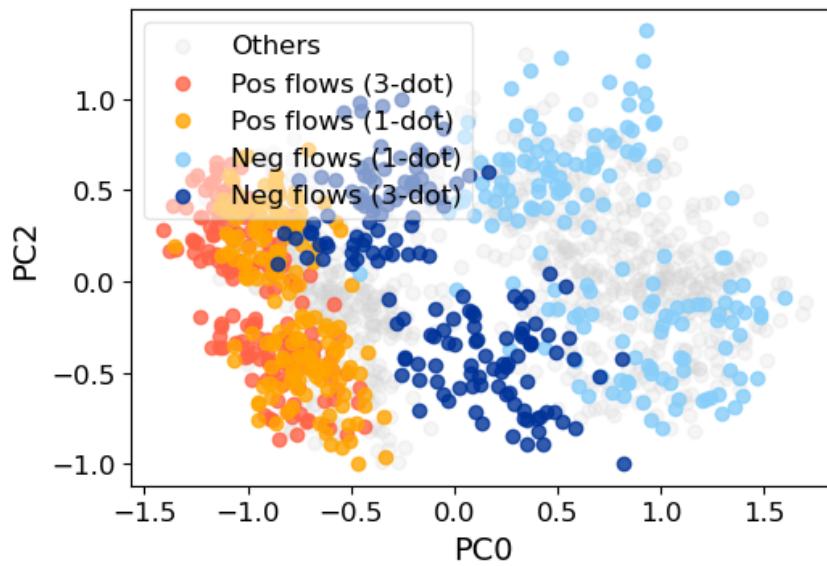
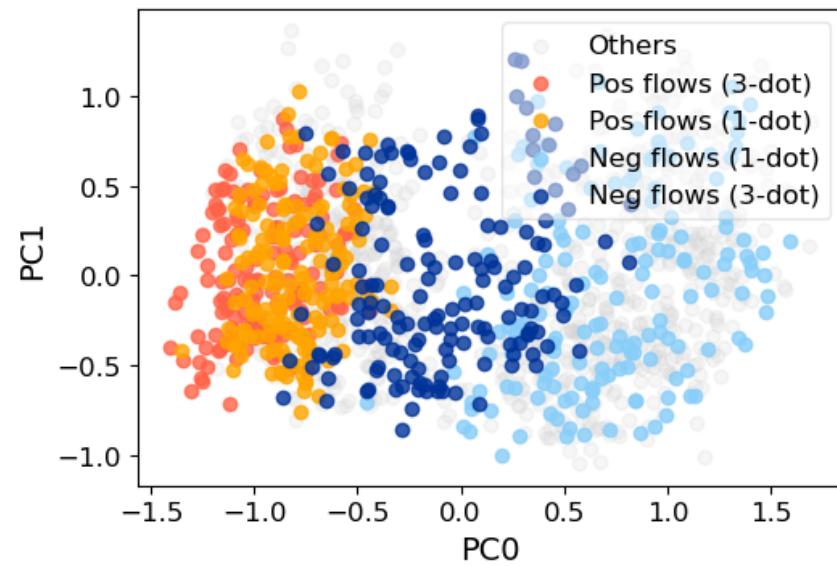
PCA: Pos vs neg flows (3-dot)



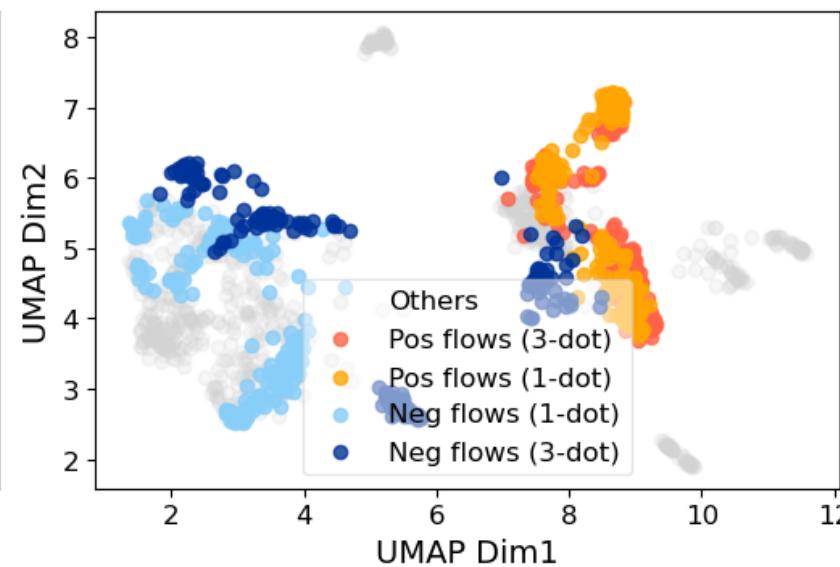
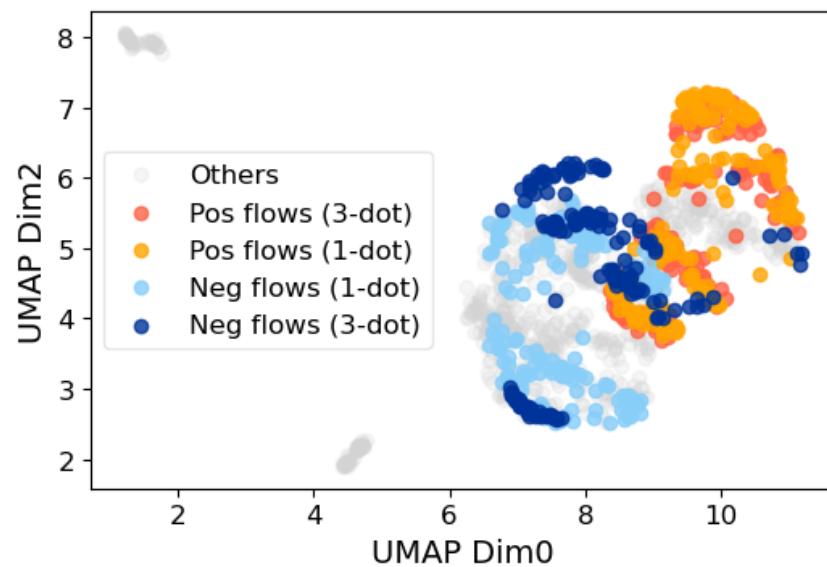
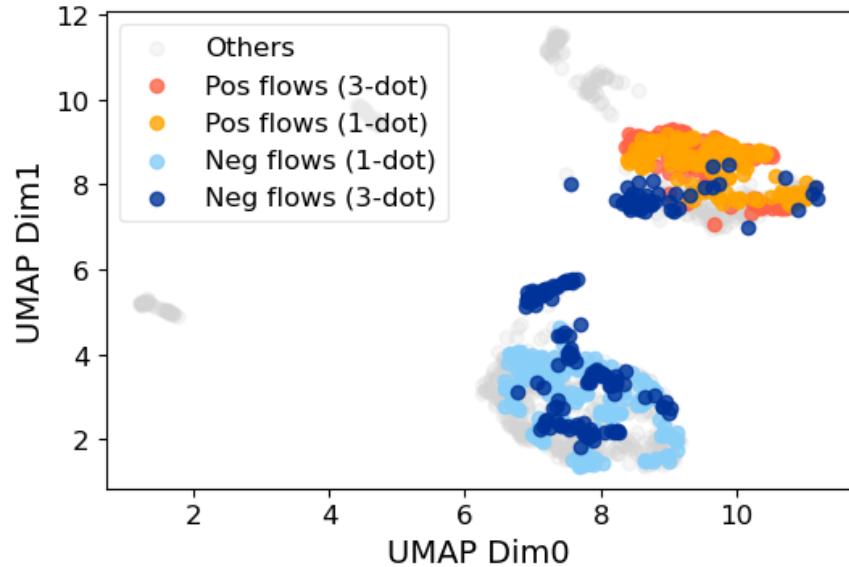
UMAP: Pos vs neg flows (3-dot)



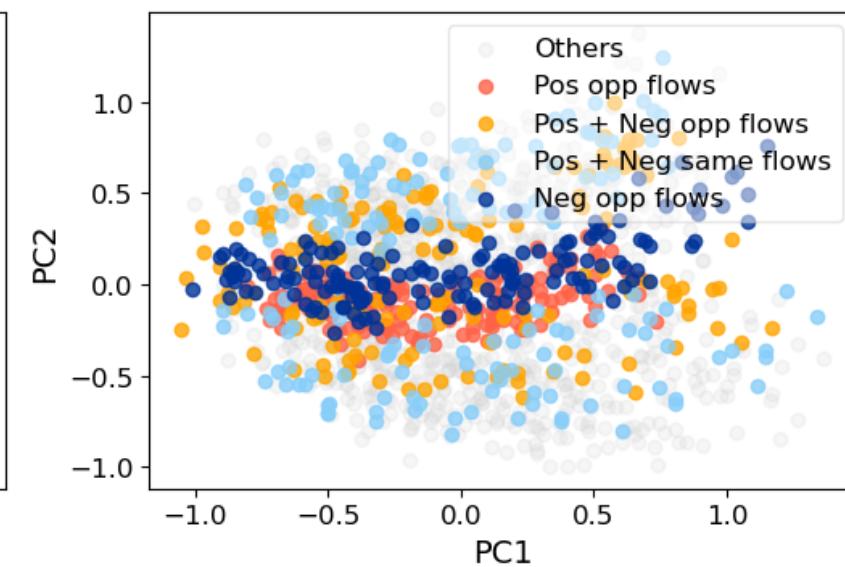
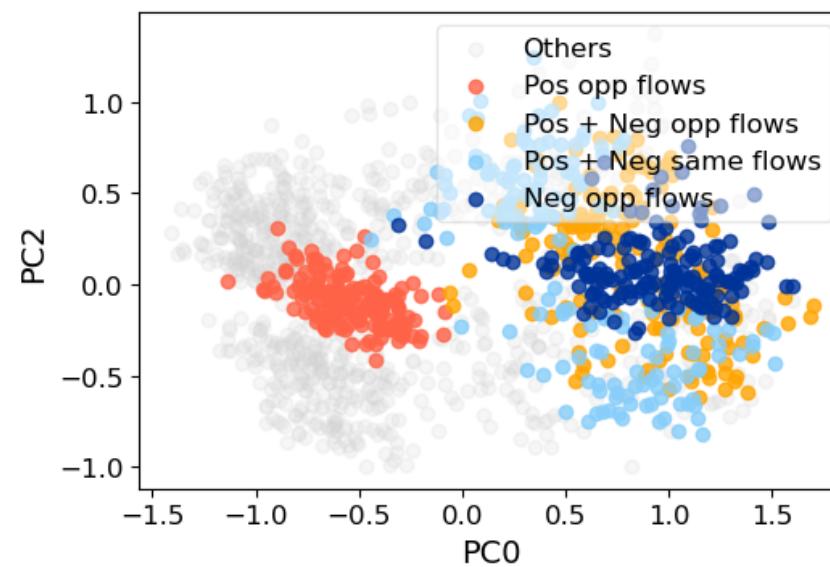
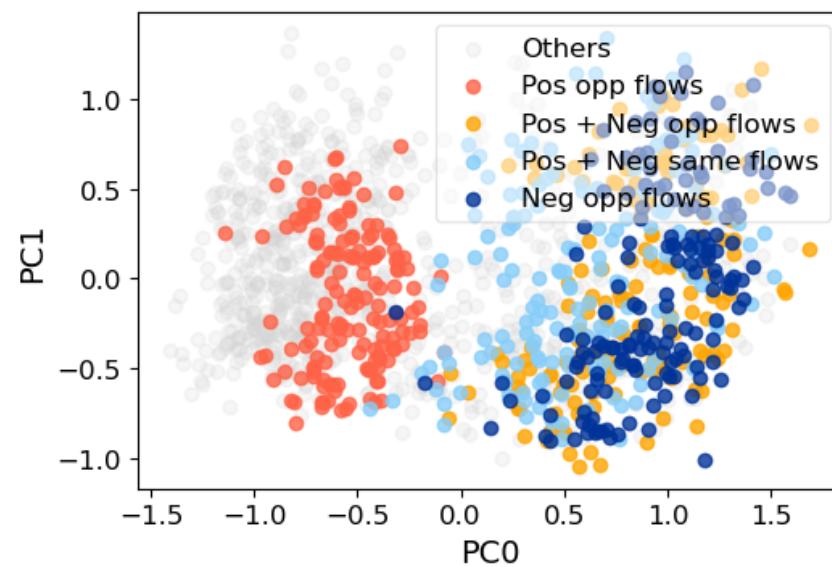
PCA: Pos vs neg flows (1 & 3-dot)



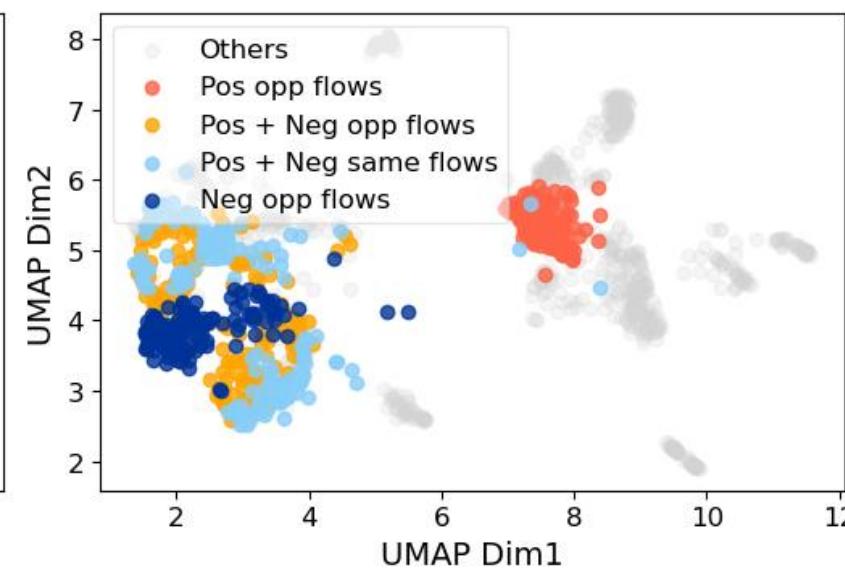
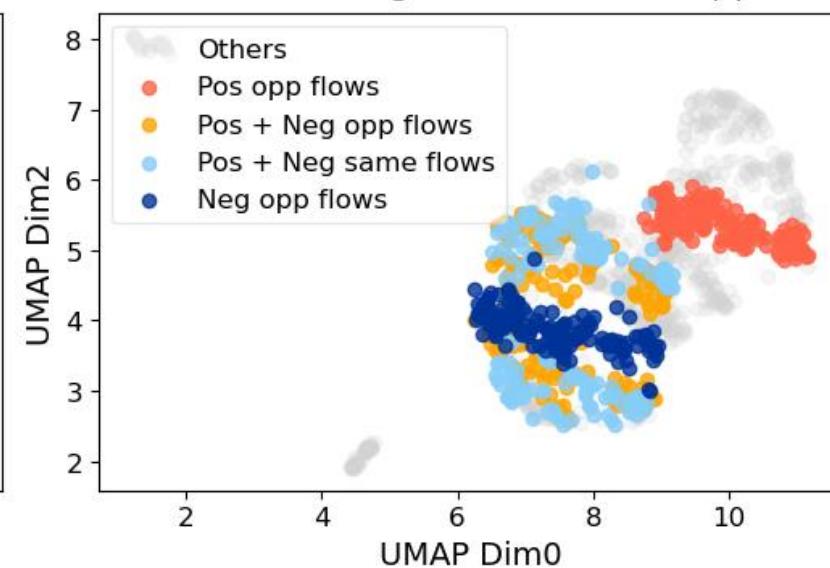
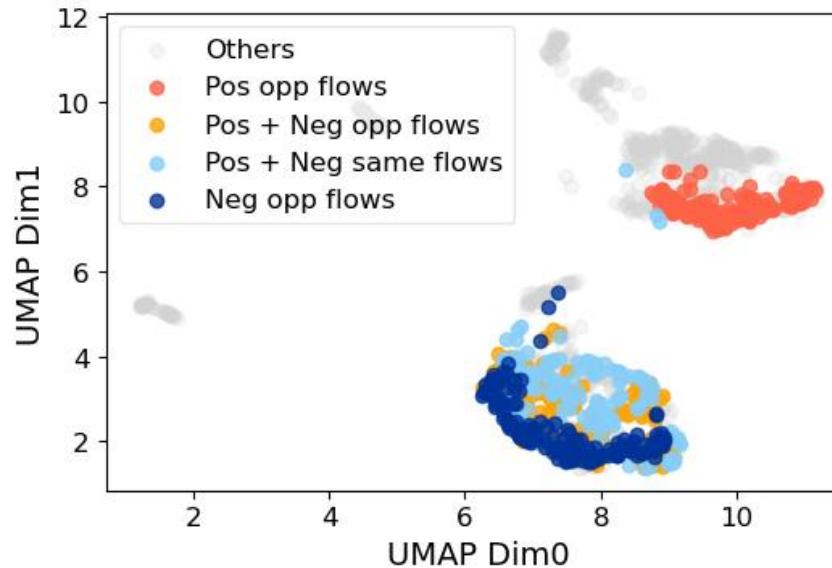
UMAP: Pos vs neg flows (1 & 3-dot)



PCA: Pos vs neg flows (same & opp)

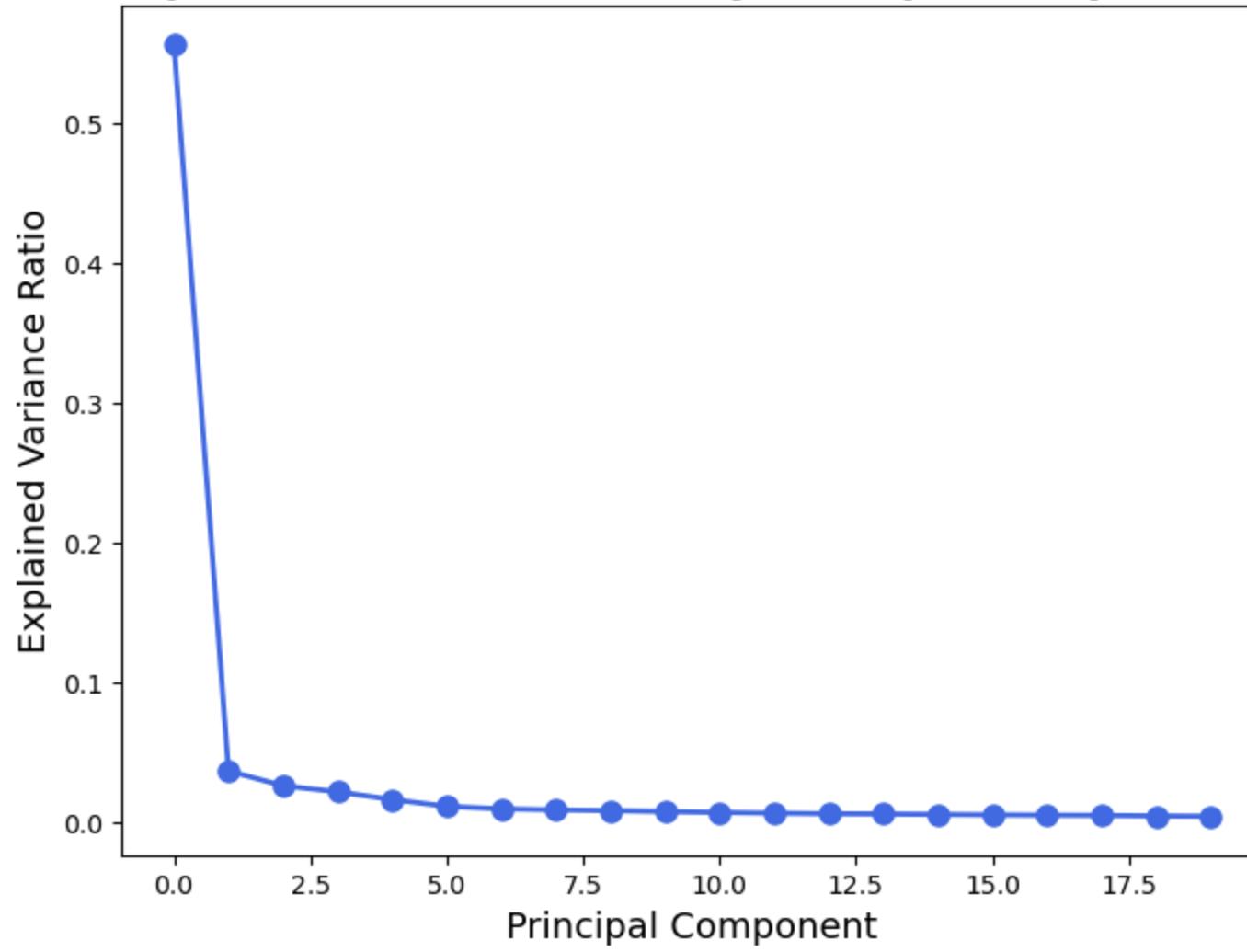


UMAP: Pos vs neg flows (same & opp)



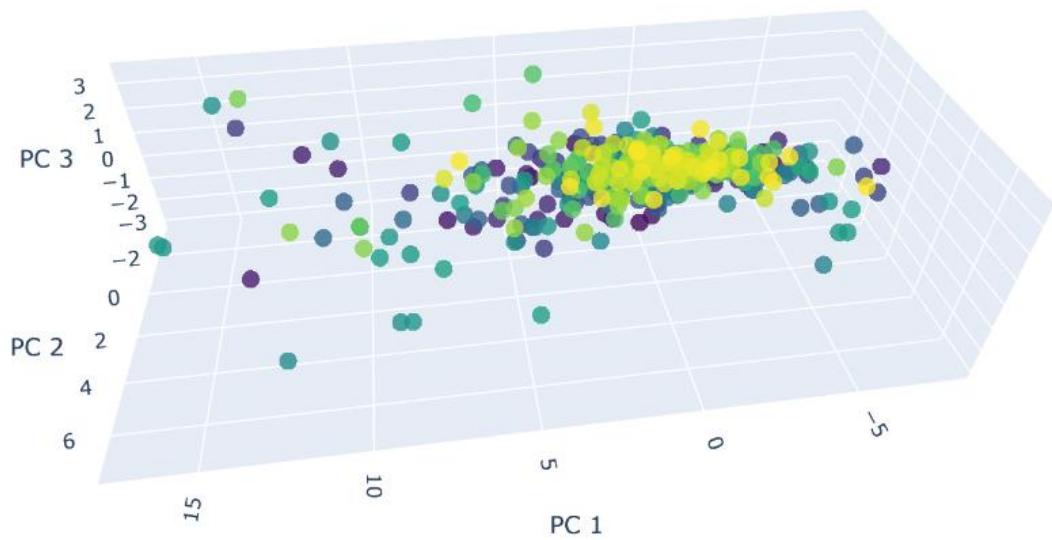
Cells in dLGN from Multiple FOVs with 36-Stim Set
PCA, UMAP, Manifold Embedding (reduce stimuli dim)

Explained Variance Ratio by Principal Components

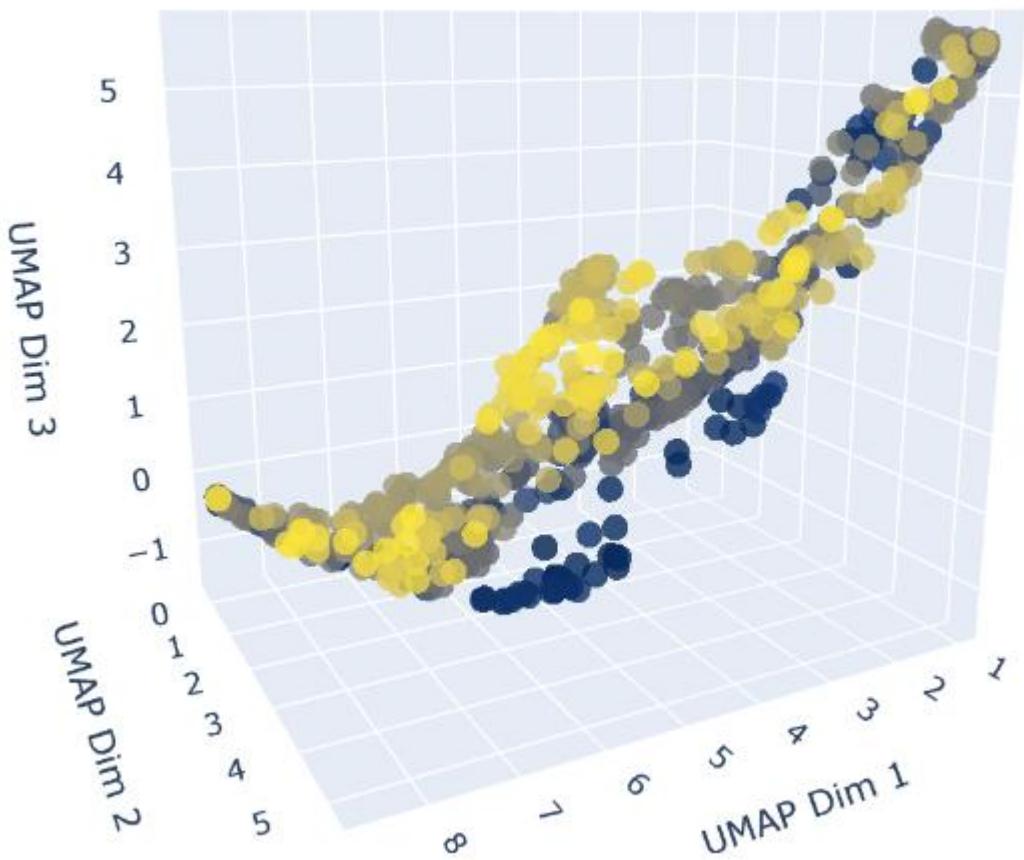


We use the first 3 PCs for plotting.

PCA: Neurons in 3D

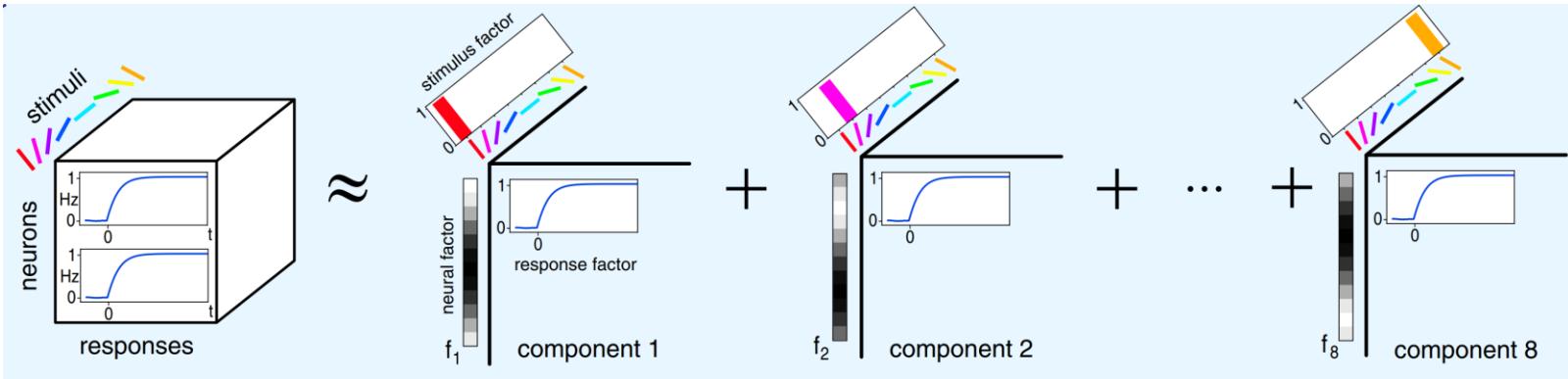


UMAP: Neurons in 3D



Manifold Embedding

Tensor Decomposition



Data here is a 3-D tensor (traces):
Neurons, Stimuli, Responses.

Neural factors: How each neuron responds to different stimuli.

Stimulus factors: The components of the stimuli that drive the neural responses.

Response factors: The temporal dynamics of the neural responses.

L. Dyballa et al 2024

- **Neuron factor matrix** $N \in \mathbb{R}^{I \times R}$,
- **Stimulus factor matrix** $S \in \mathbb{R}^{J \times R}$,
- **Response factor matrix** $R \in \mathbb{R}^{K \times R}$,

where R is the rank of the decomposition (i.e., the number of hidden components), and the number of columns R represents the number of latent factors.

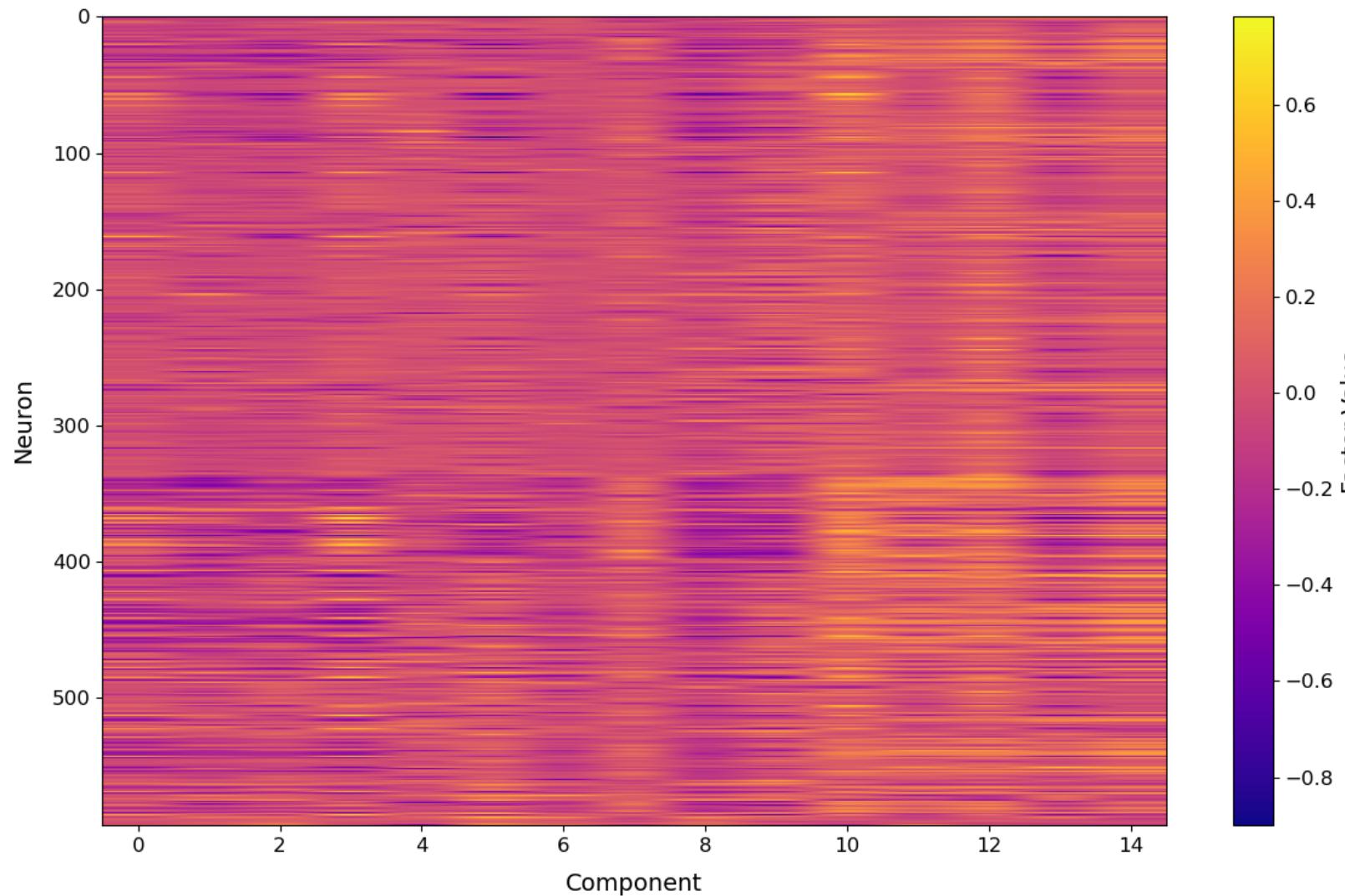
$$\min_{N,S,R} \frac{1}{2} \|T - \sum_{r=1}^R N(:, r) \circ S(:, r) \circ R(:, r)\|_F^2$$

Iterative Optimization:

Since the problem is non-convex and involves multiple variables, an **iterative optimization** approach is used. A common method is **Alternating Least Squares (ALS)** or **gradient-based methods**. Here's the general idea:

- **Initialization:** Start by initializing the factor matrices N , S , and R randomly with non-negative values.
- **Iterative Updates:**
 - Fix S and R , then solve for N by minimizing the objective with respect to N .
 - Fix N and R , then solve for S .
 - Fix N and S , then solve for R .

Neuron Factor Matrix

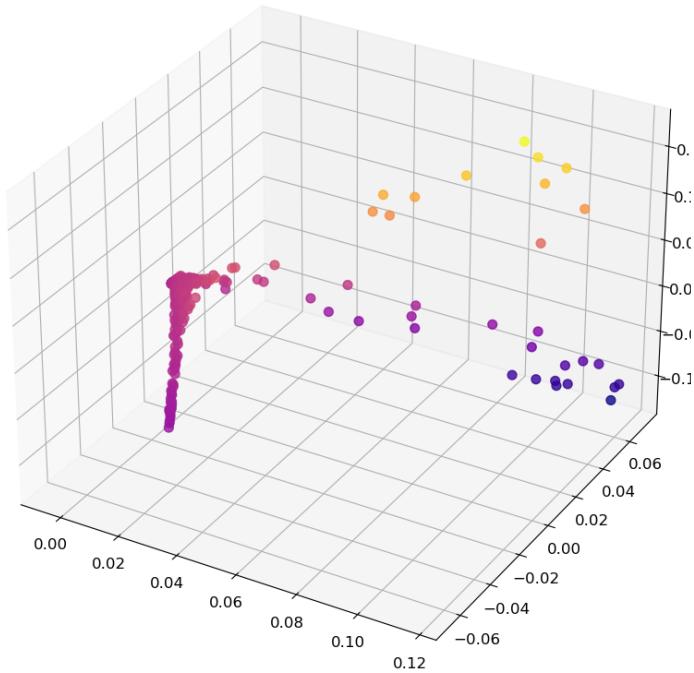


Use the neuron factor matrix to calculate pairwise distance.

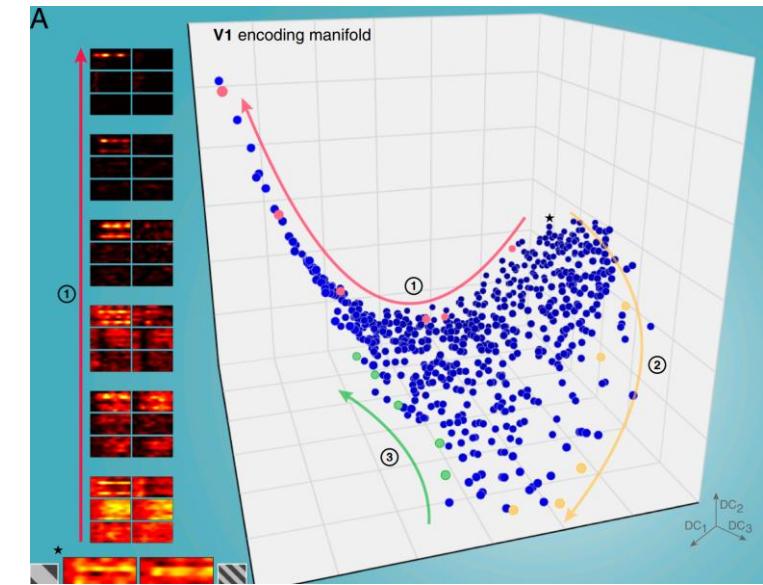
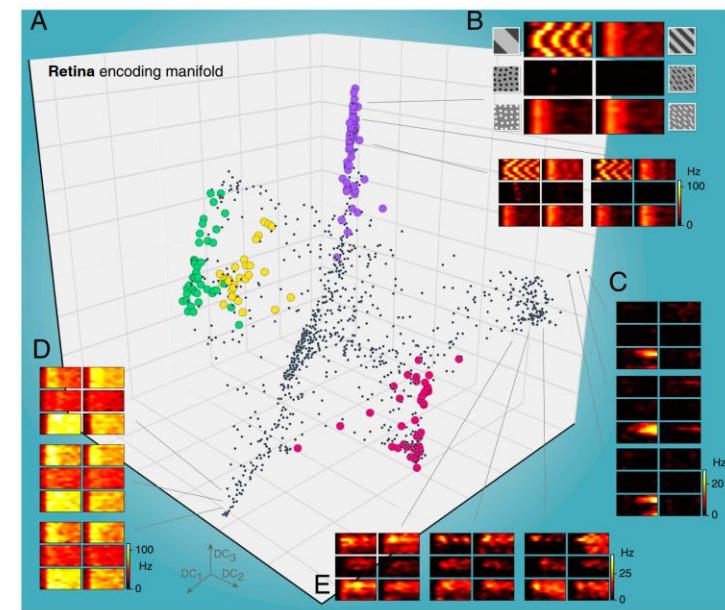
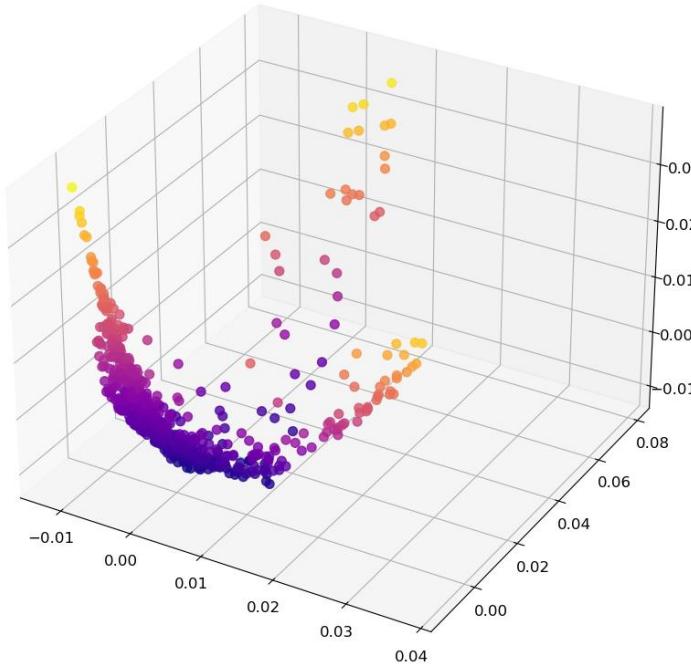
Compute Iterated Adaptive Neighborhoods (IAN kernel), where nonlinearity comes from, from pairwise distances, obtaining data graphs (unweighted and weighted)

1. Compute the similarity matrix D .
2. Normalize the similarity matrix to obtain the **diffusion matrix P** .
3. Compute the eigenvectors and eigenvalues of the diffusion matrix.
4. Use the top eigenvectors as coordinates in the low-dimensional space, generating the **manifold embedding** of neurons.

Neuron Diffusion Map (Weighted)



Neuron Diffusion Map (Unweighted)

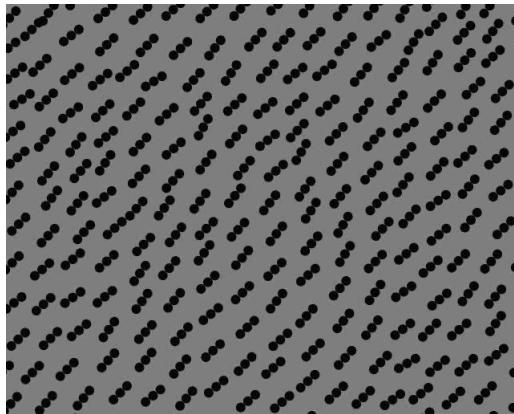


Cells in dLGN from Multiple FOVs with 48-Stim Set
PCA & UMAP (reduce neurons dim)

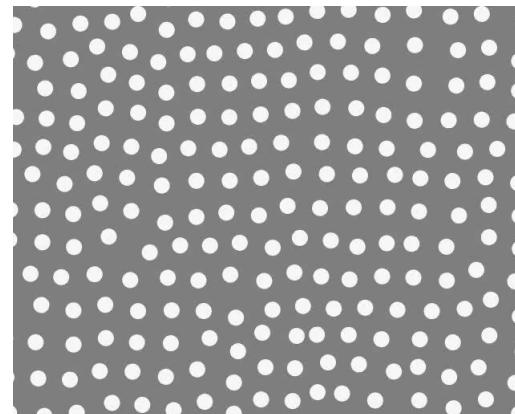
Stimuli

```
# 0, 45, 90, 135, 180, 225, 270, 315
gratings_lf
gratings_hf
neg_1flows
pos_1flows
neg_3flows
pos_3flows
```

```
# 48 stimuli
label_names = [
    "neg_3flows_45", "gratings_lf_90", "gratings_hf_45", "neg_1flows_270",
    "pos_1flows_135", "pos_3flows_180", "pos_1flows_45", "neg_3flows_315",
    "gratings_hf_270", "gratings_lf_0", "pos_3flows_45", "neg_1flows_45",
    "neg_1flows_0", "pos_1flows_0", "pos_3flows_0", "gratings_lf_225",
    "gratings_hf_90", "neg_3flows_180", "gratings_hf_0", "neg_1flows_135",
    "pos_3flows_225", "pos_1flows_315", "neg_3flows_135", "gratings_lf_180",
    "pos_3flows_90", "gratings_lf_135", "pos_1flows_180", "neg_3flows_0",
    "gratings_hf_225", "neg_1flows_180", "gratings_hf_135", "pos_3flows_315",
    "pos_1flows_225", "gratings_lf_270", "neg_1flows_225", "neg_3flows_270",
    "neg_1flows_315", "gratings_hf_180", "pos_3flows_270", "neg_3flows_225",
    "pos_1flows_270", "gratings_lf_45", "pos_1flows_90", "neg_1flows_90",
    "gratings_hf_315", "gratings_lf_315", "pos_3flows_135", "neg_3flows_90"
]
```



neg_3flows_315

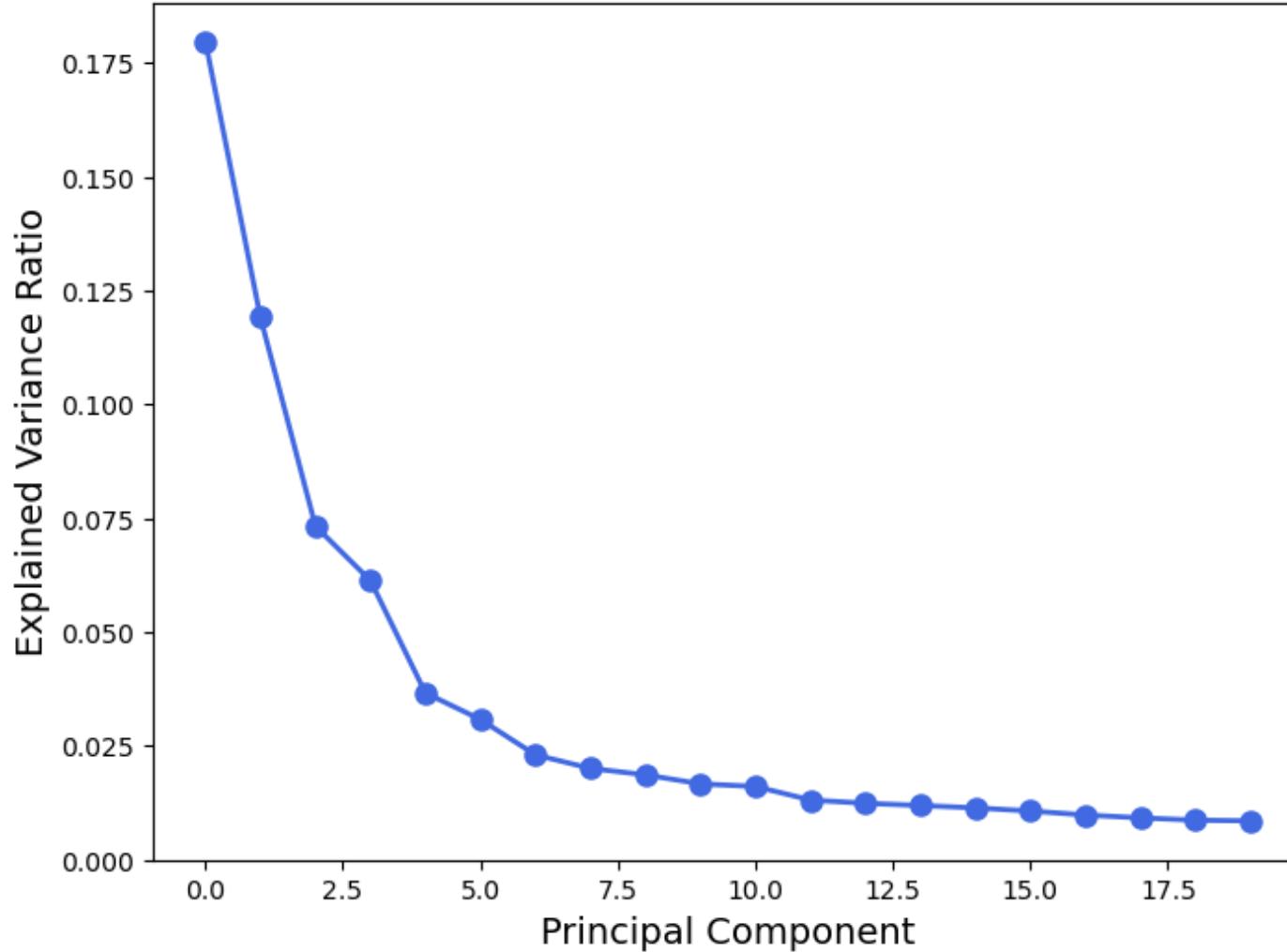


pos_1flows_225



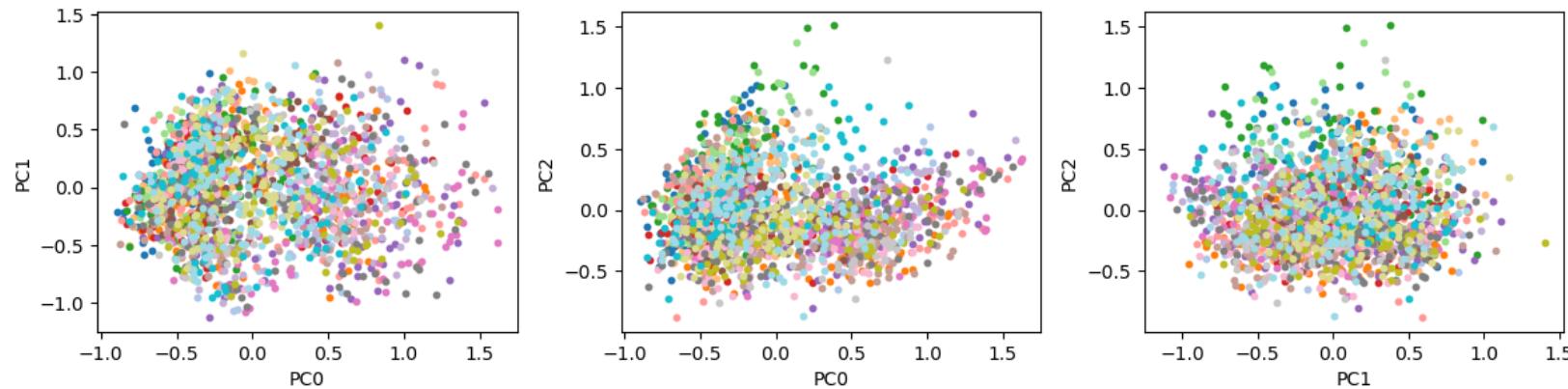
gratings_lf_45

Explained Variance Ratio by Principal Components

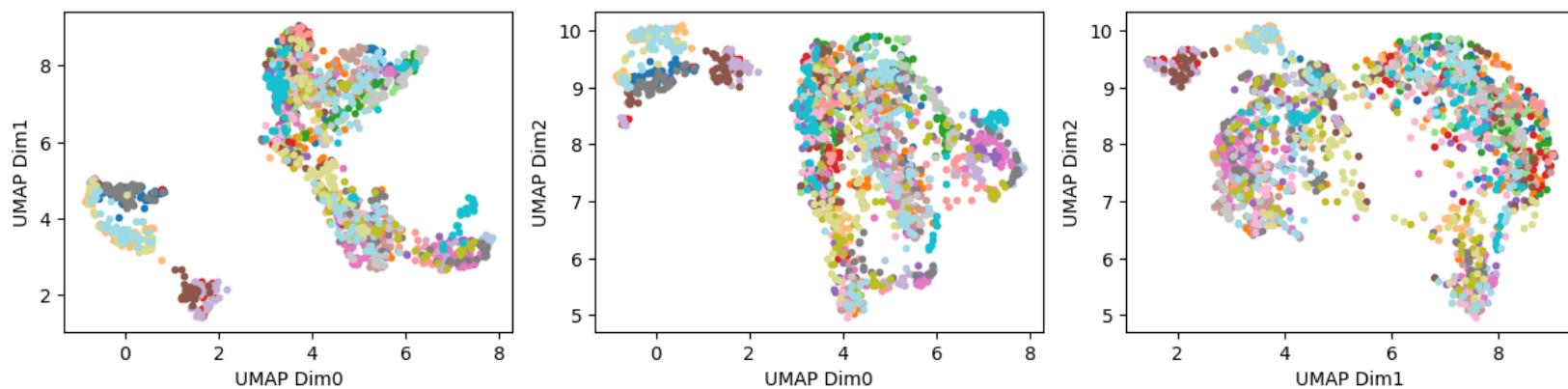


We use the first 3 PCs for plotting.

PCA: All Stimuli with Labels

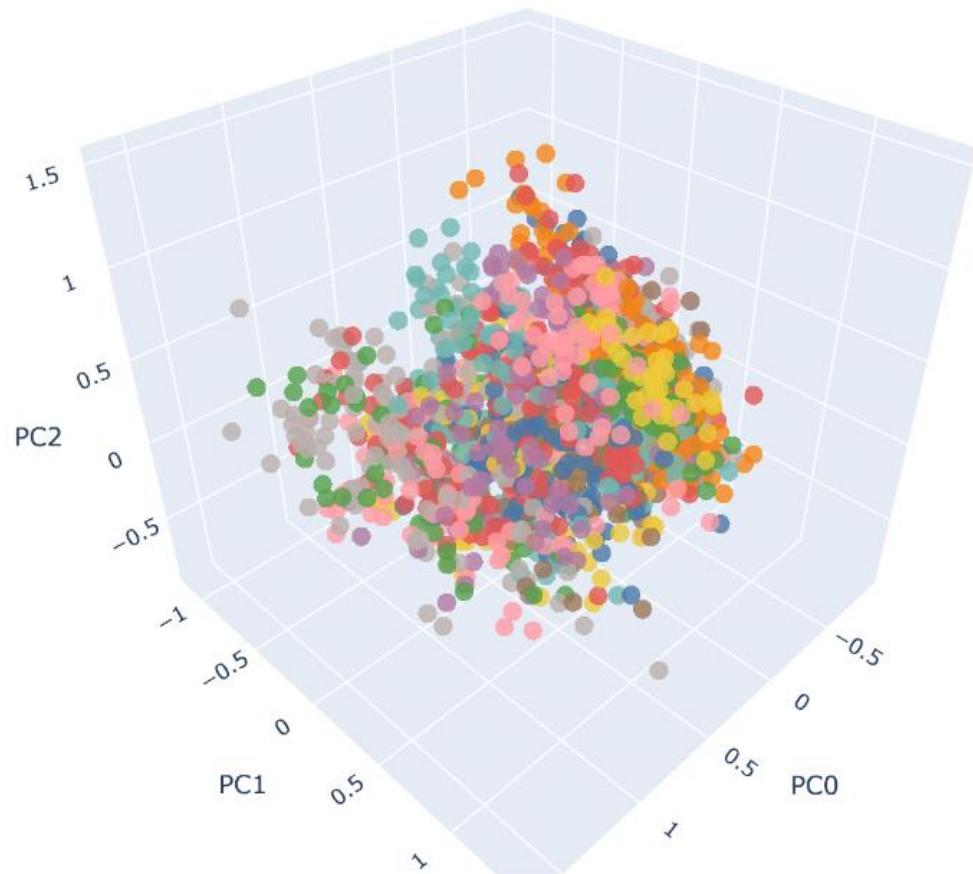


UMAP: All Stimuli with Labels

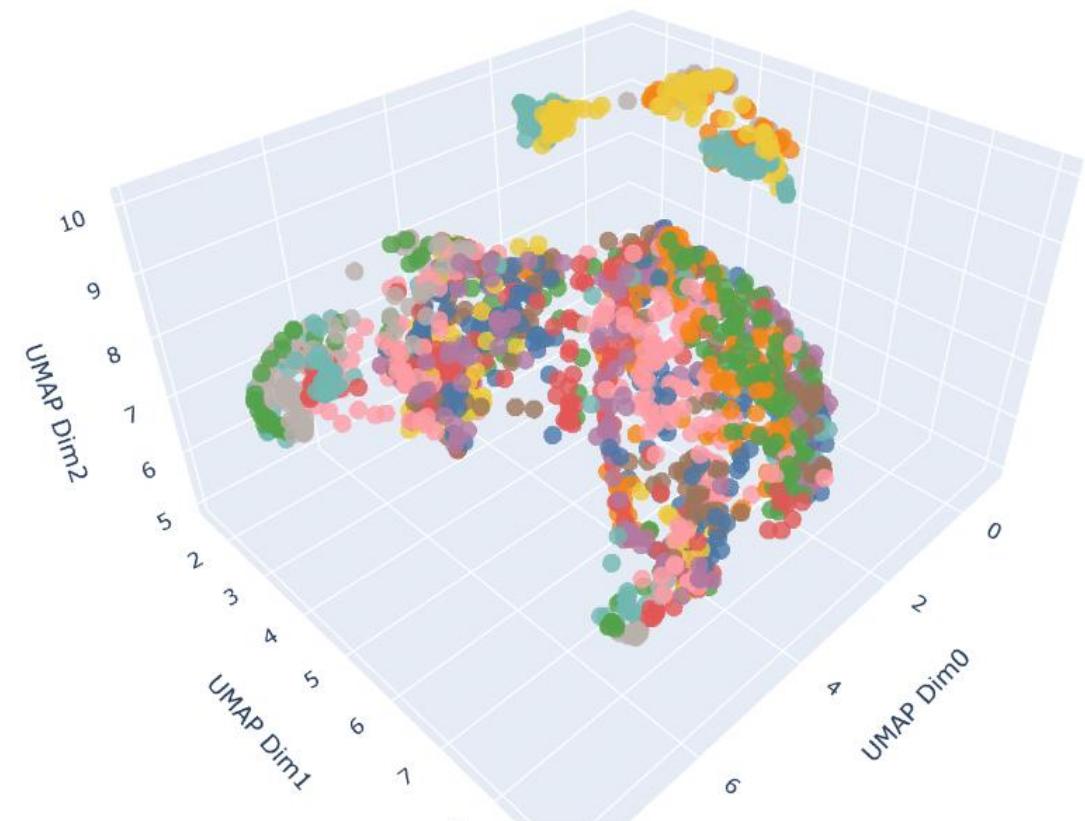


- neg_3flows_45
 - pos_1flows_45
 - neg_1flows_0
 - gratings_hf_0
 - gratings_lf_180
 - gratings_hf_225
 - gratings_lf_270
 - pos_3flows_270
 - neg_1flows_90
 - gratings_lf_90
 - neg_3flows_315
 - pos_1flows_0
 - neg_1flows_135
 - pos_3flows_90
 - neg_1flows_180
 - neg_1flows_225
 - neg_3flows_225
 - neg_3flows_270
 - gratings_hf_45
 - gratings_hf_270
 - pos_3flows_0
 - pos_3flows_225
 - gratings_lf_135
 - gratings_hf_135
 - neg_3flows_270
 - pos_1flows_270
 - neg_1flows_270
 - neg_3flows_0
 - neg_1flows_225
 - neg_1flows_270
 - neg_3flows_225
 - neg_3flows_270
 - pos_1flows_135
 - pos_3flows_45
 - gratings_lf_0
 - gratings_lf_225
 - pos_1flows_315
 - pos_3flows_315
 - neg_1flows_180
 - pos_3flows_315
 - neg_1flows_315
 - neg_3flows_315
 - pos_3flows_180
 - neg_1flows_45
 - neg_3flows_180
 - neg_3flows_0
 - pos_1flows_225
 - neg_1flows_180
 - neg_3flows_180

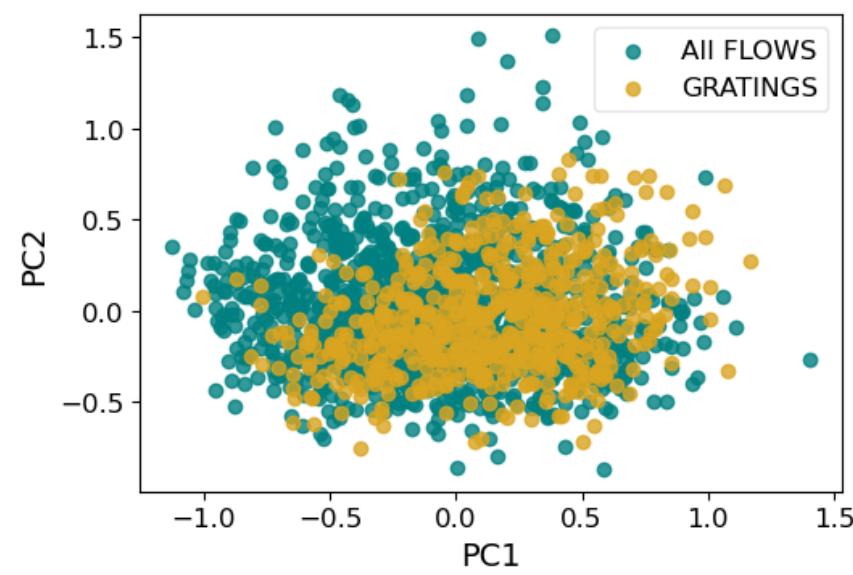
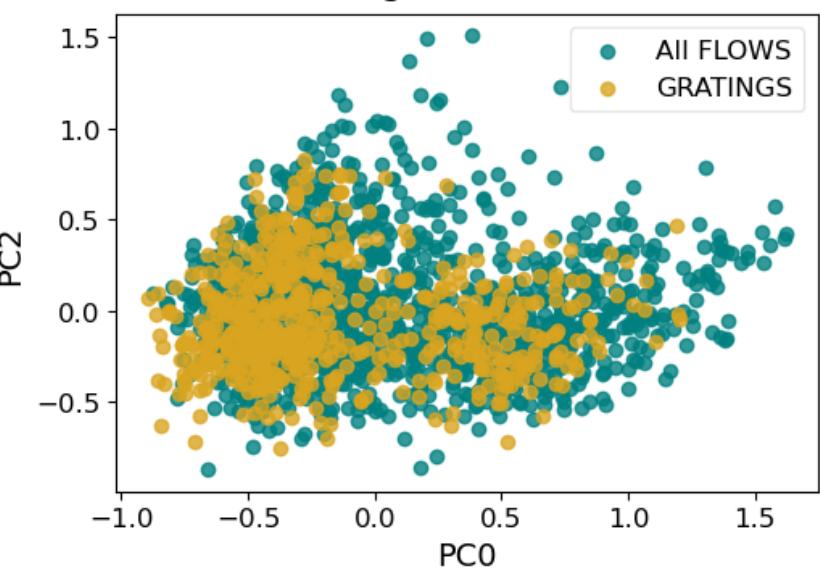
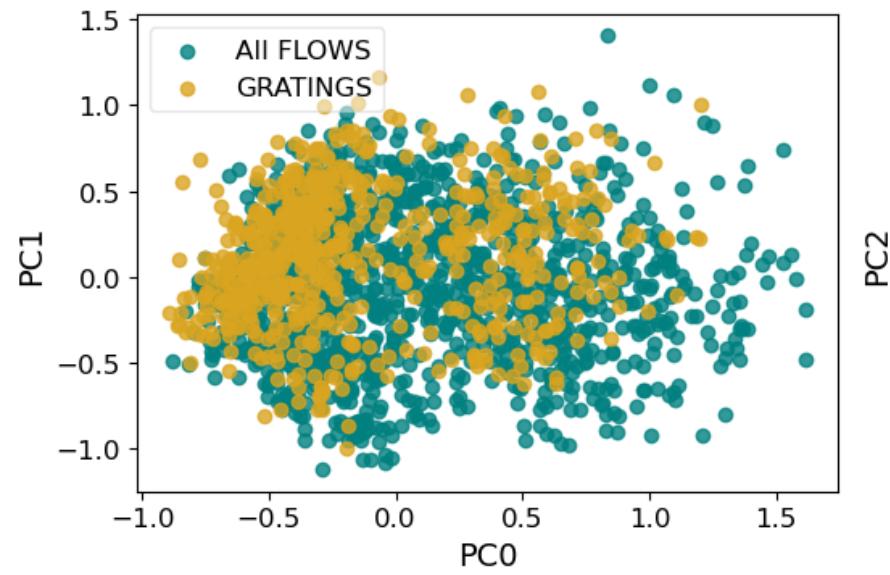
PCA: Interactive 3D Plot of All Stimuli



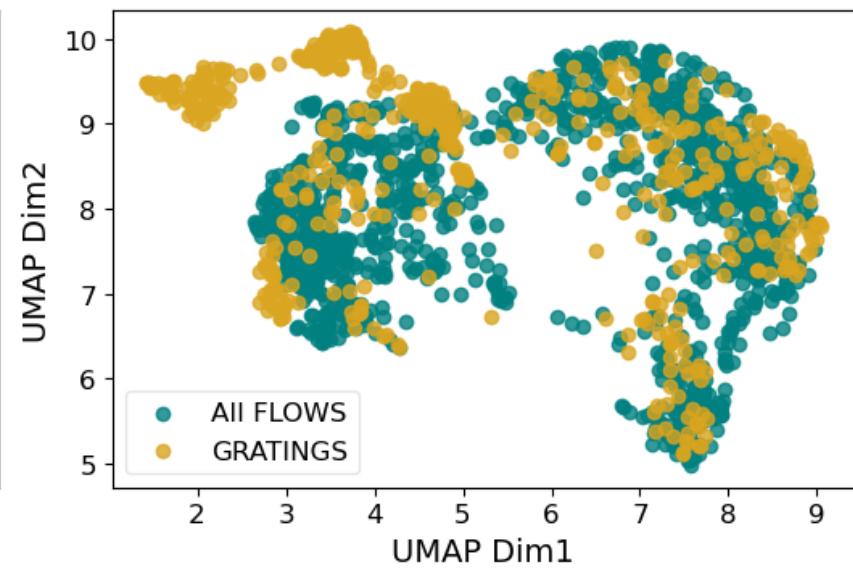
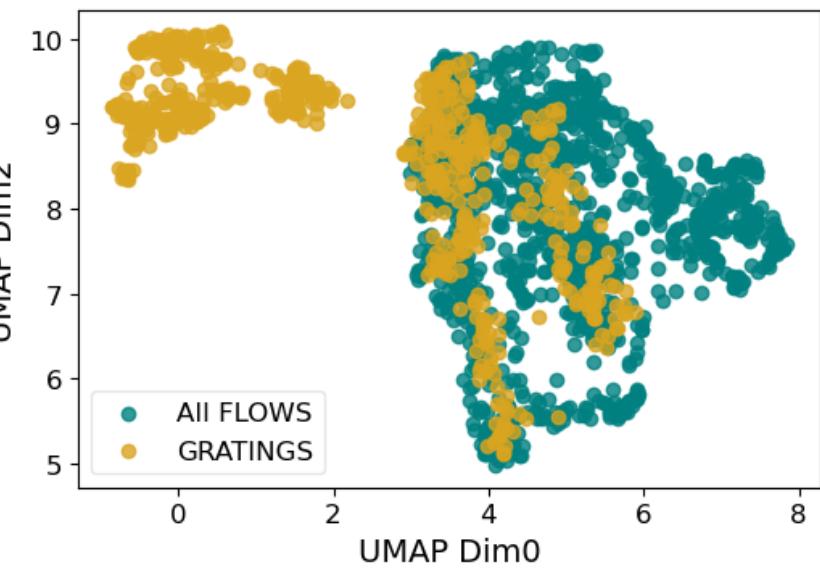
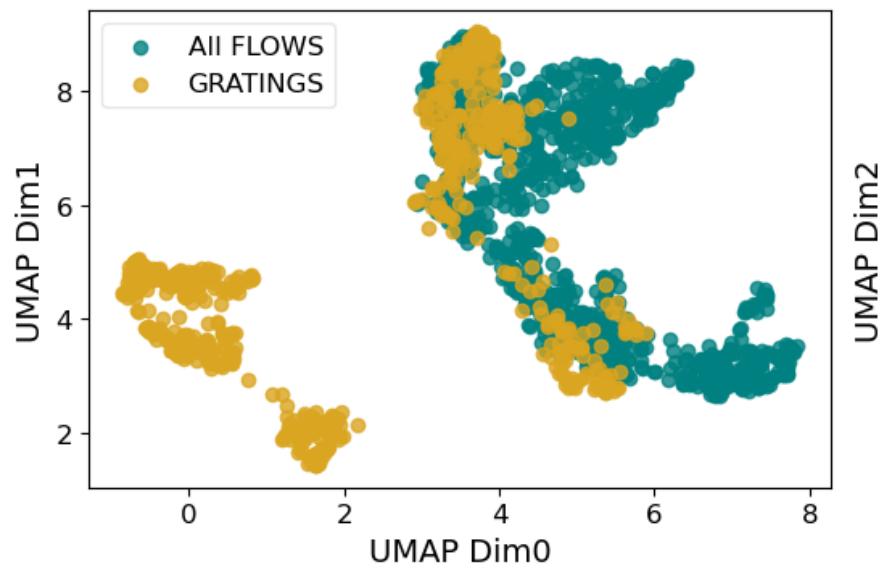
UMAP: Interactive 3D Plot of All Stimuli



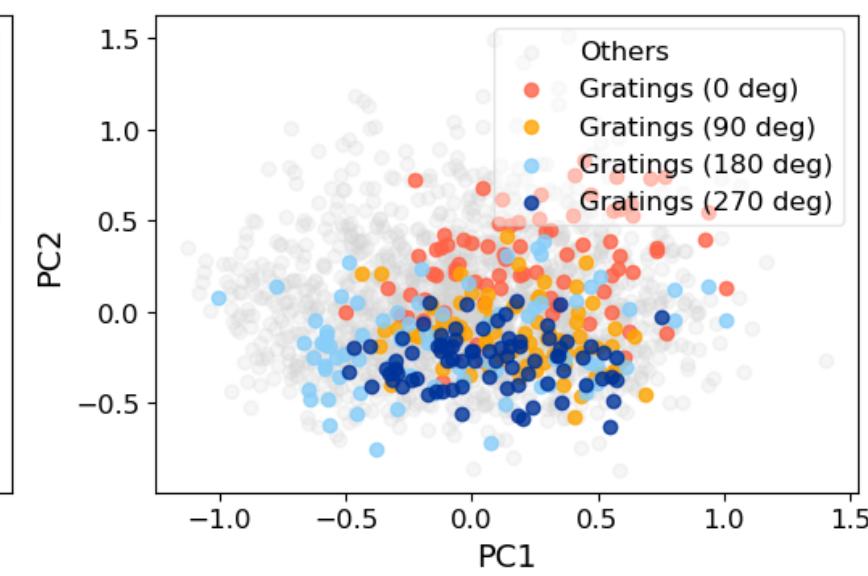
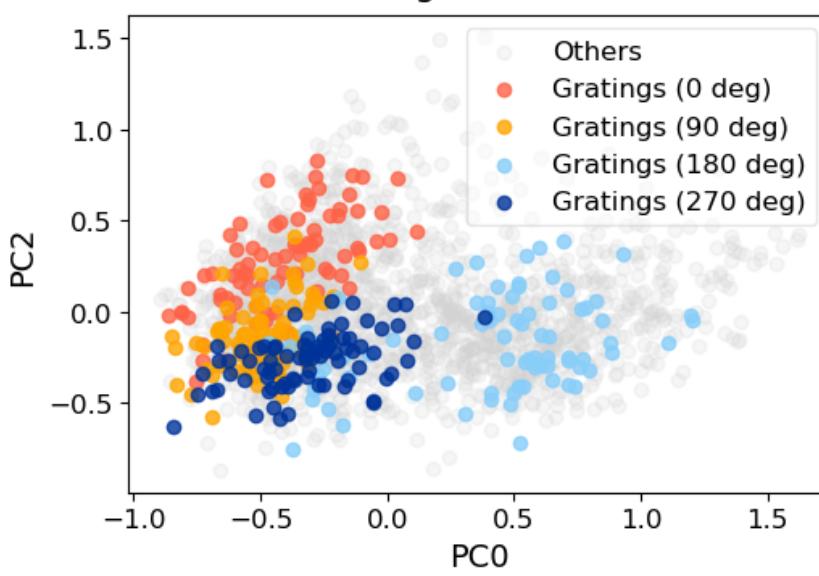
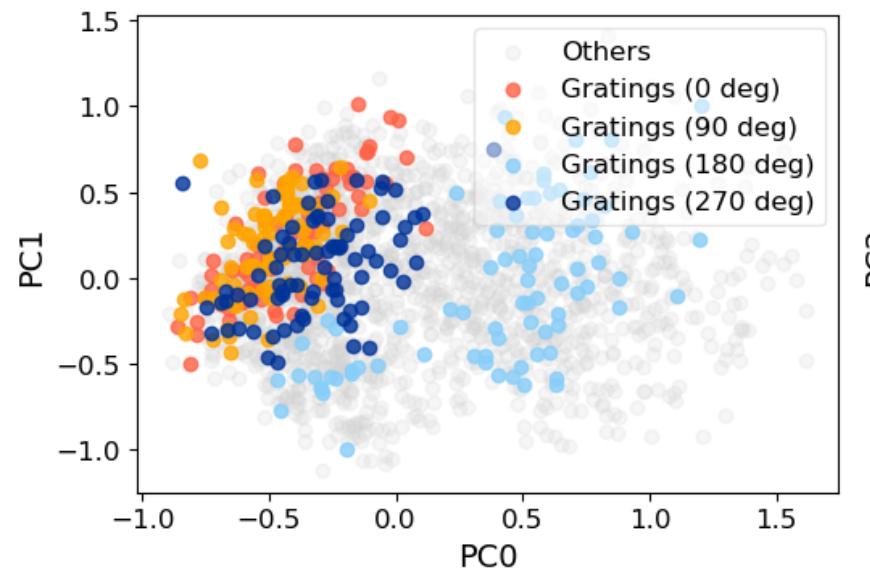
PCA: Gratings vs. All Flows



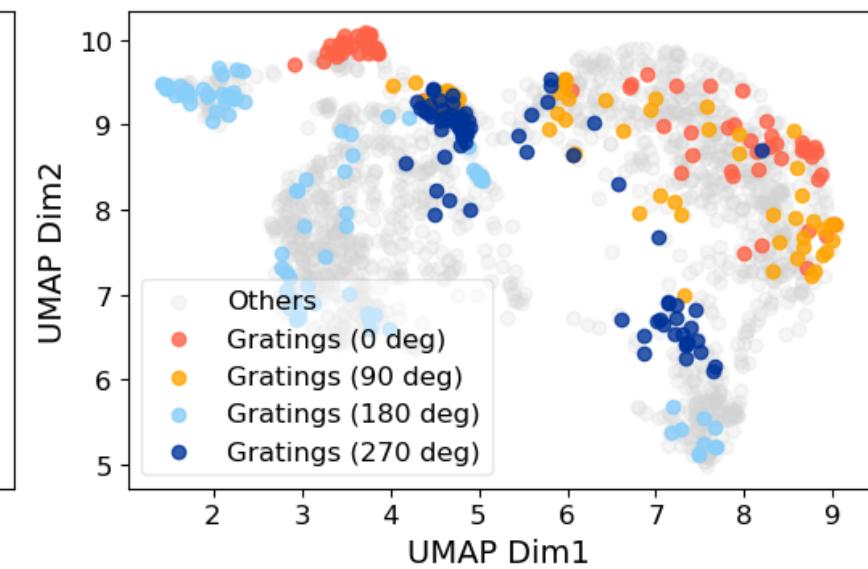
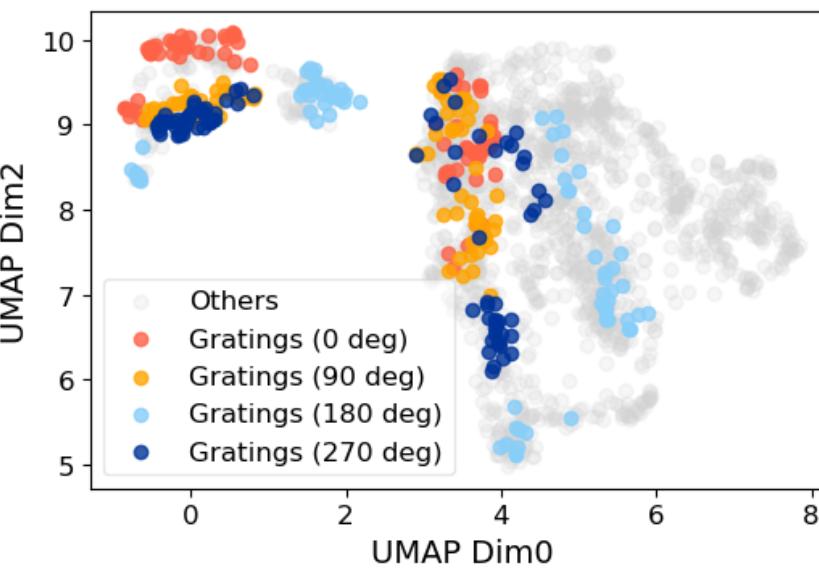
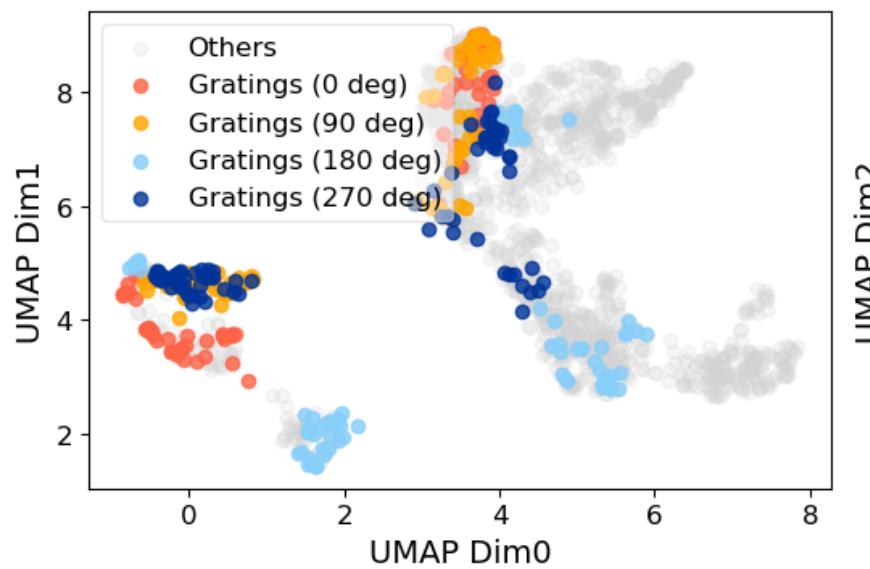
UMAP: Gratings vs. All Flows

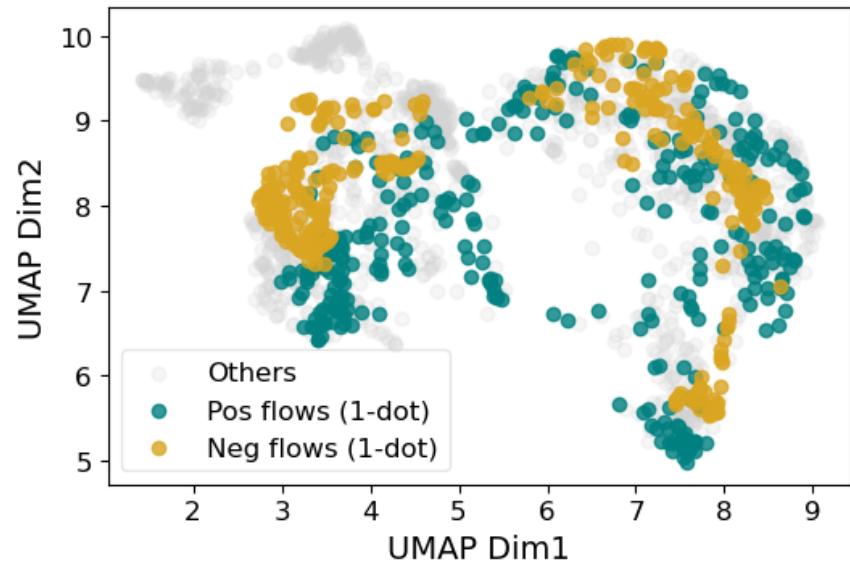
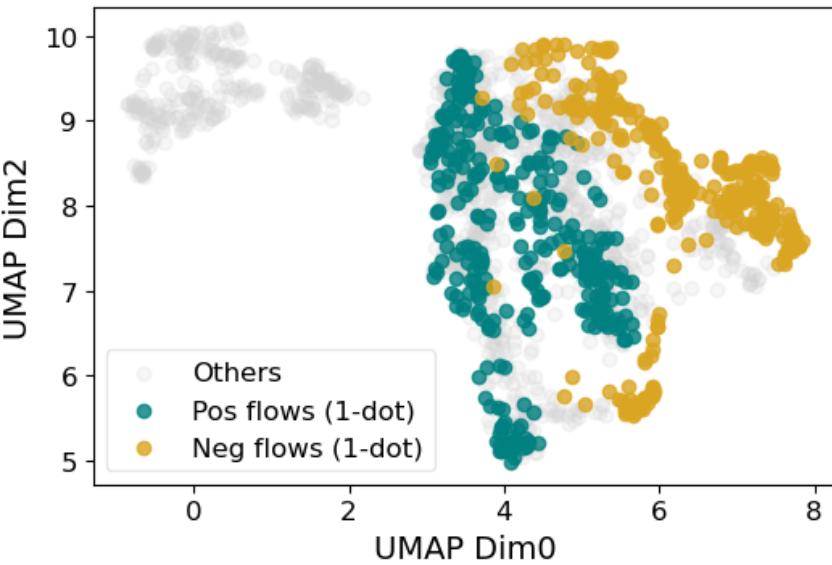
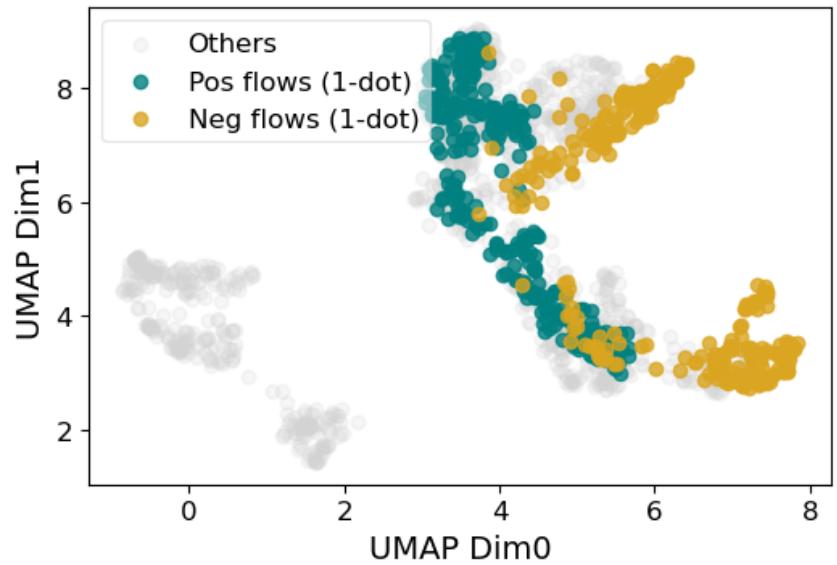
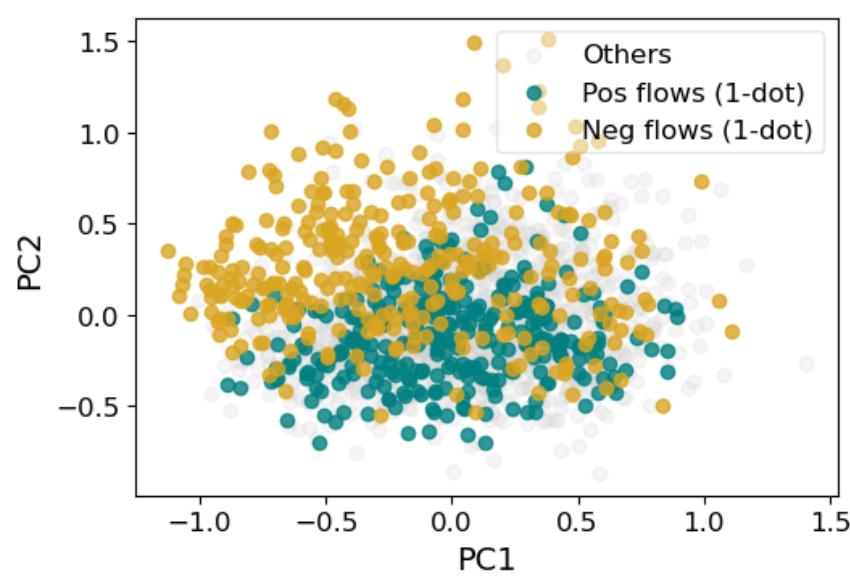
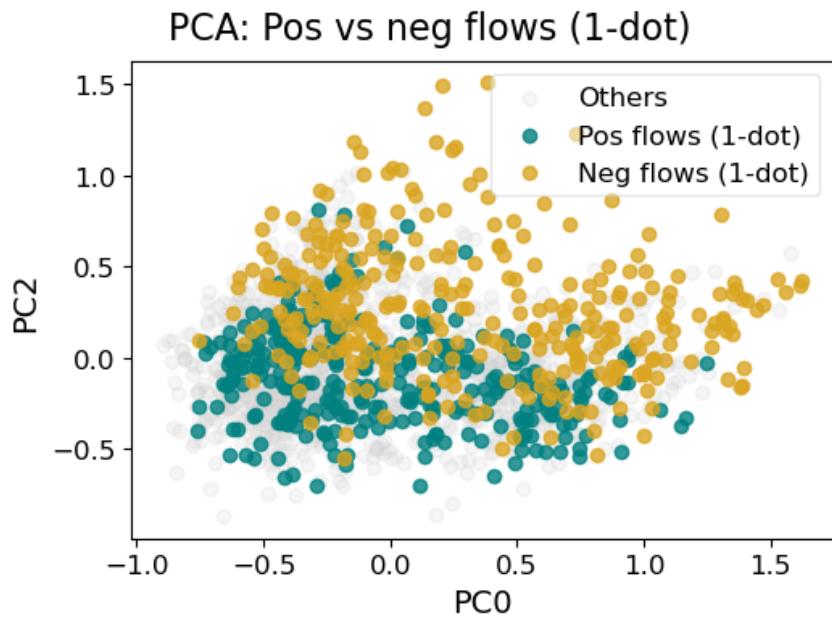
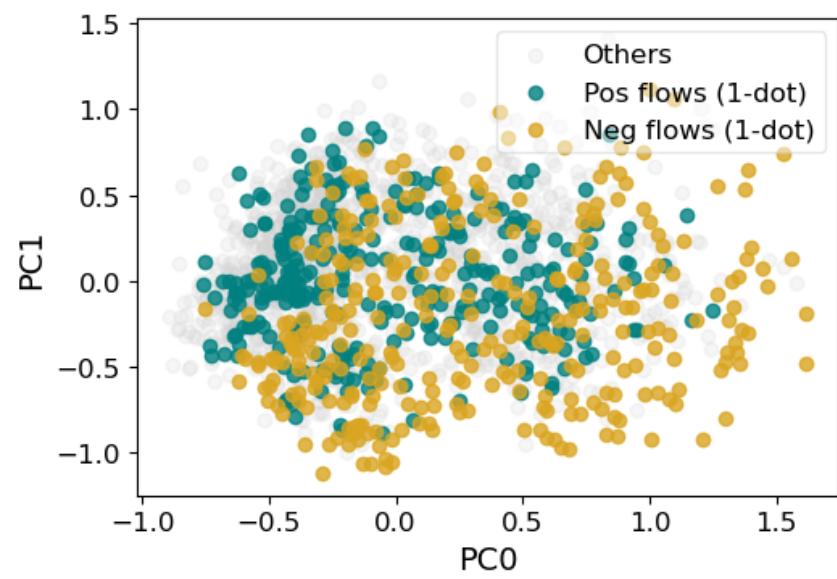


PCA: Gratings directions

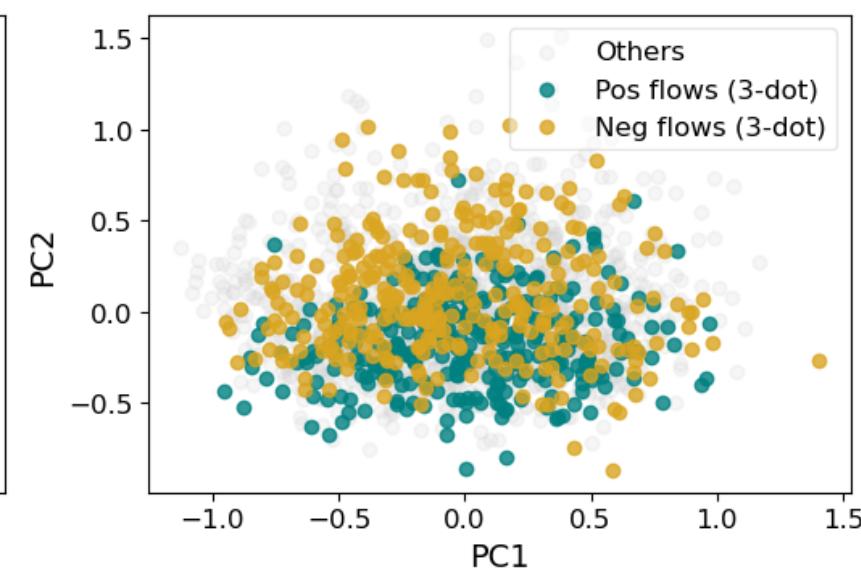
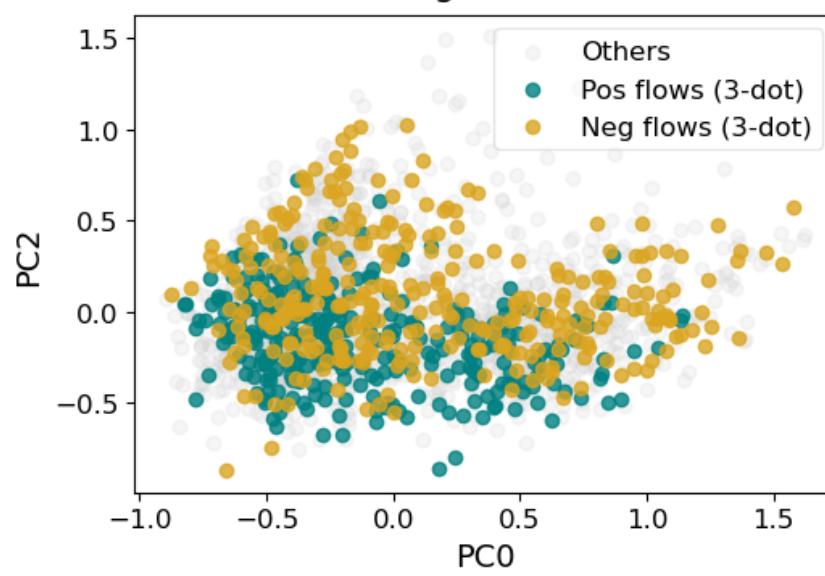
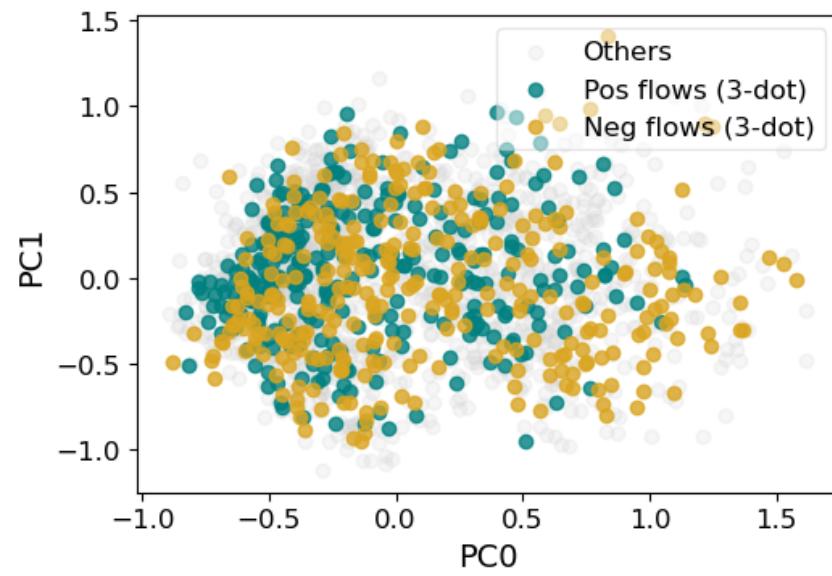


UMAP: Gratings directions

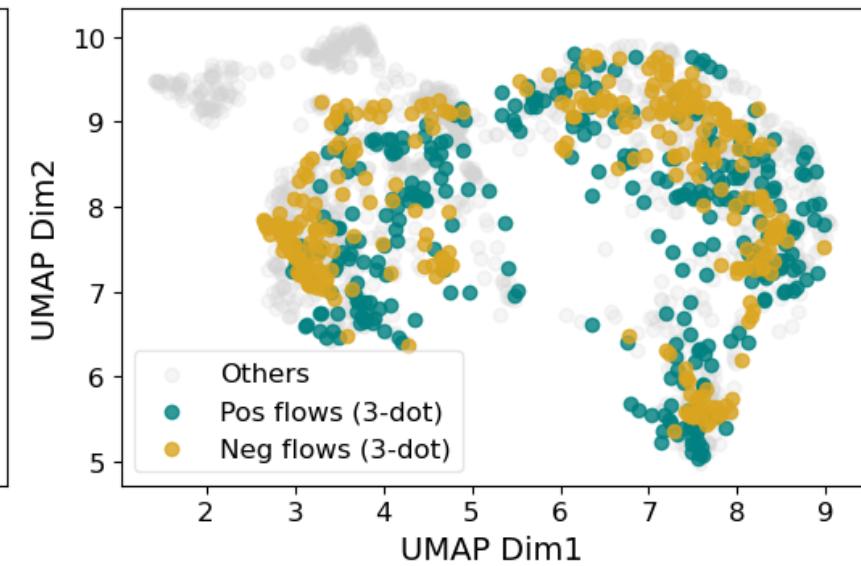
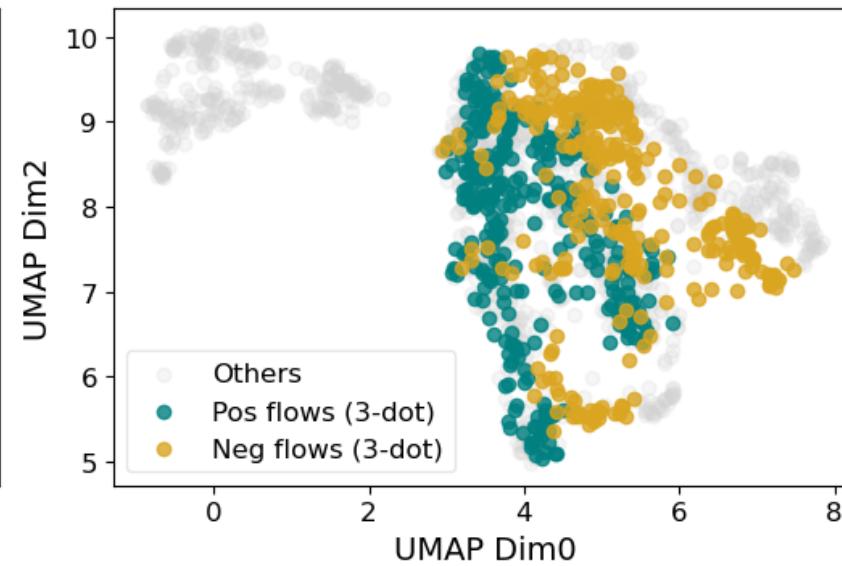
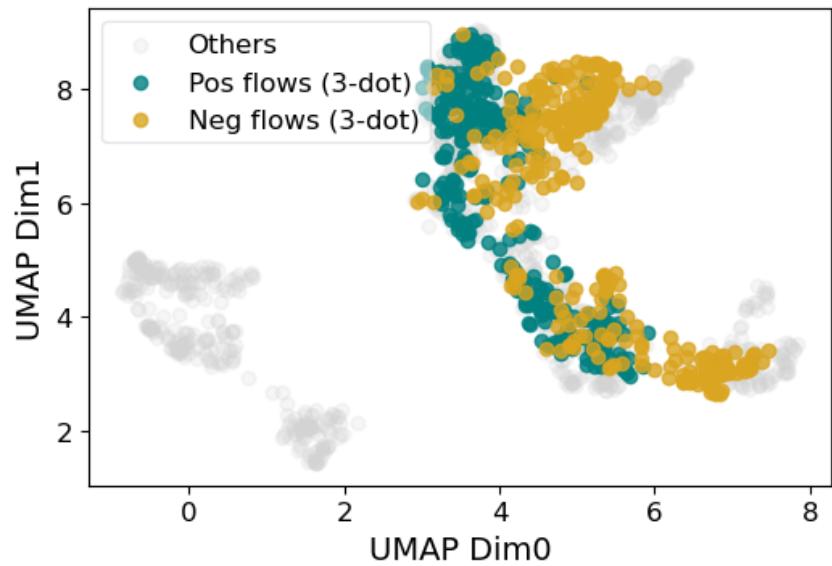




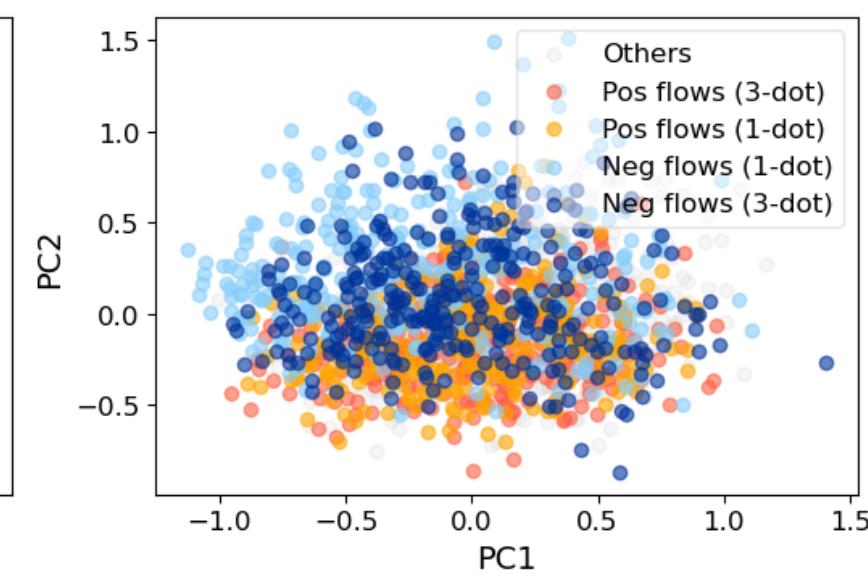
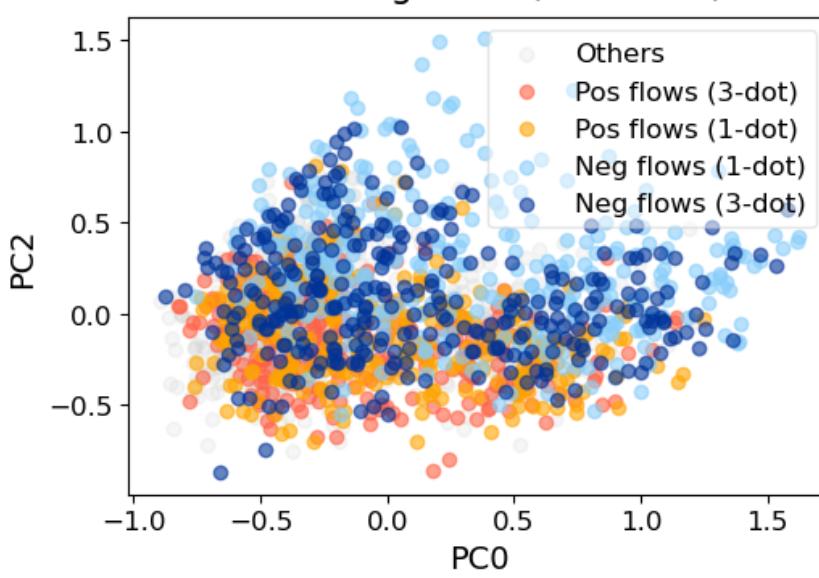
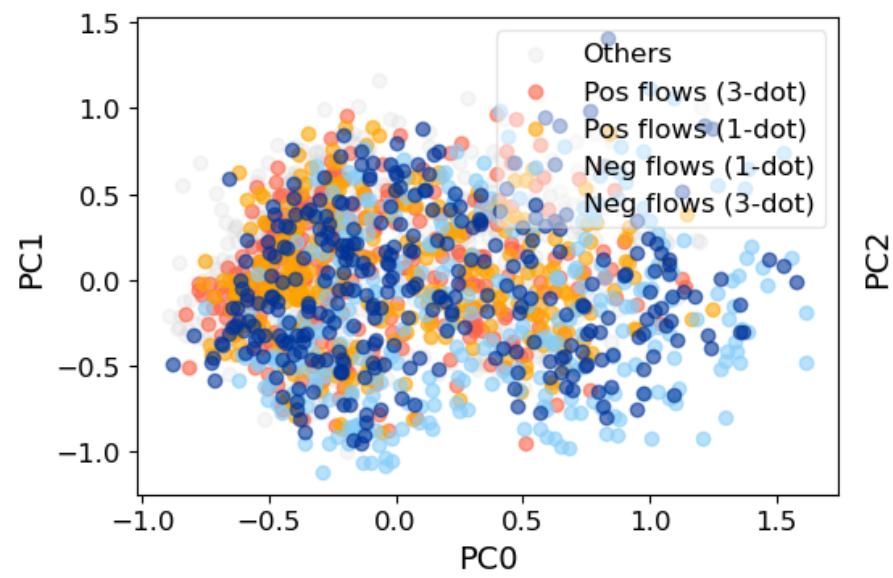
PCA: Pos vs neg flows (3-dot)



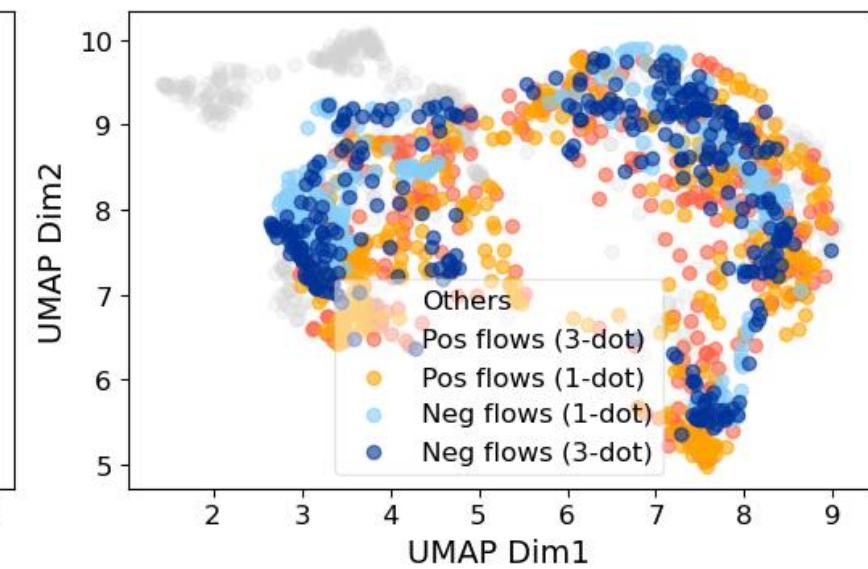
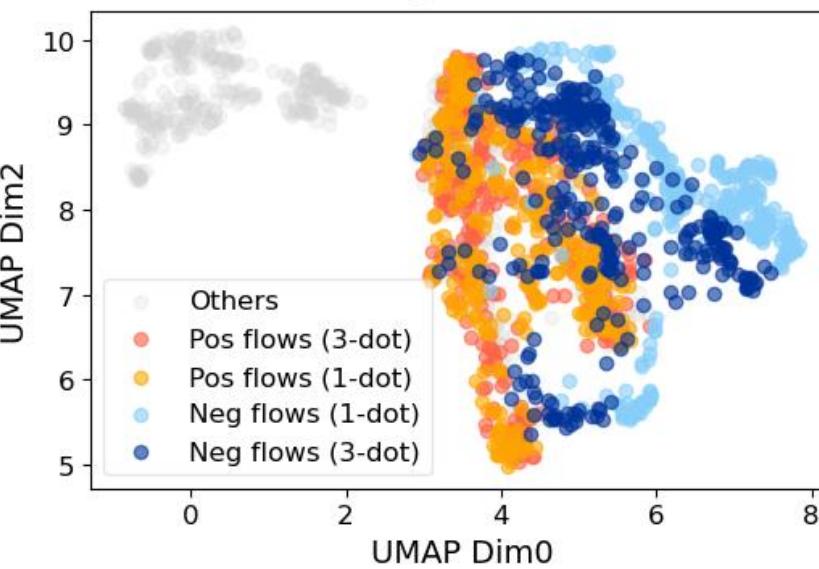
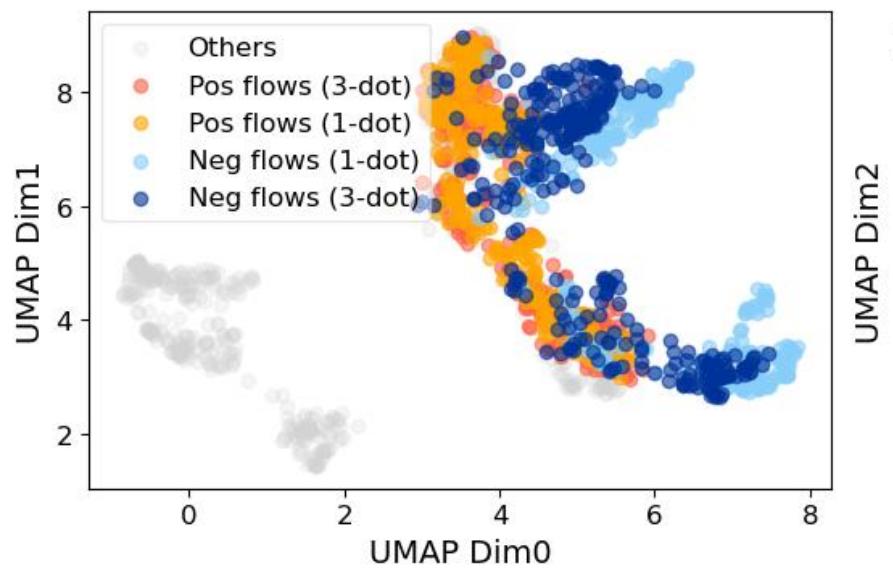
UMAP: Pos vs neg flows (3-dot)



PCA: Pos vs neg flows (1 & 3-dot)

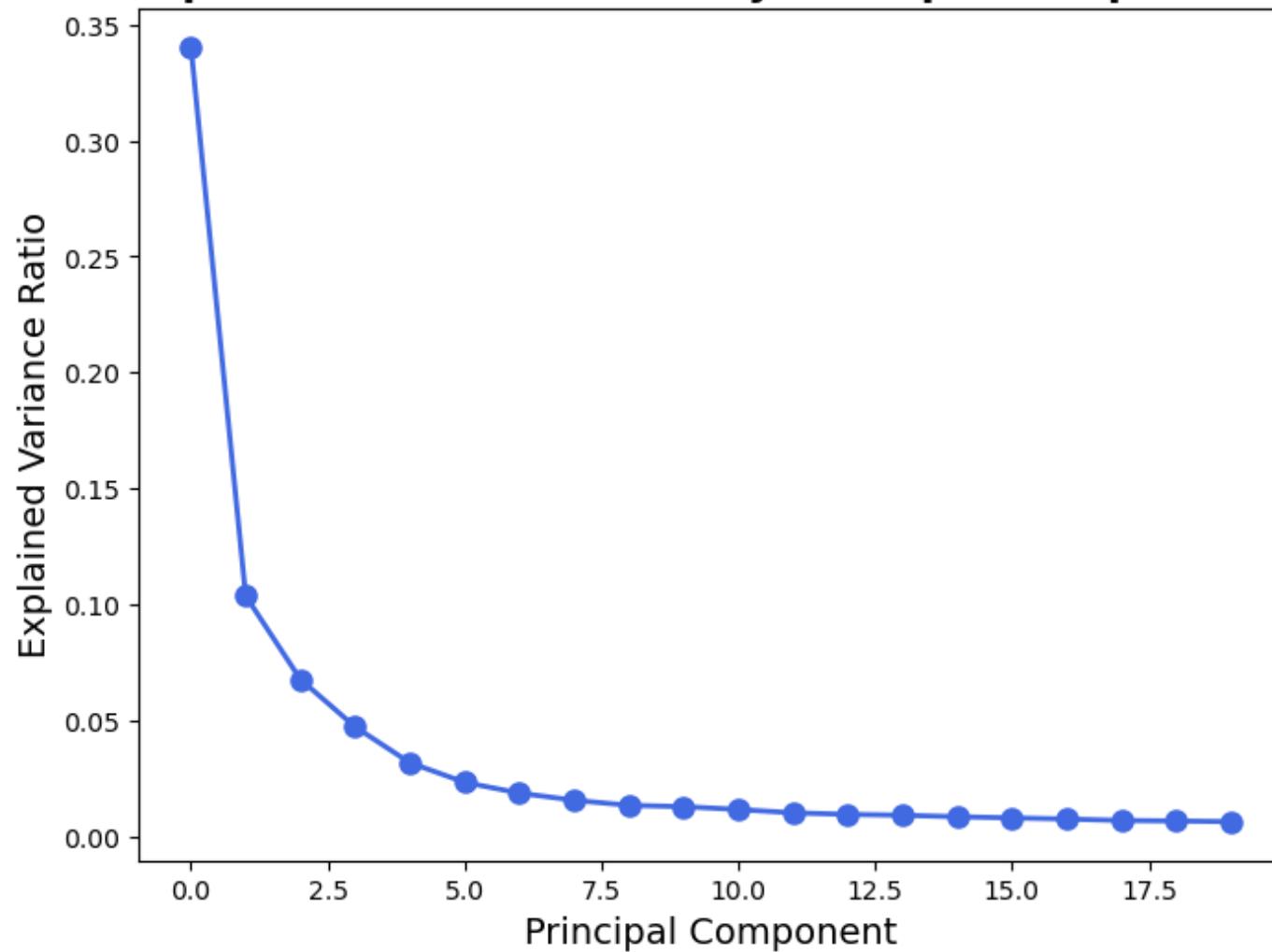


UMAP: Pos vs neg flows (1 & 3-dot)



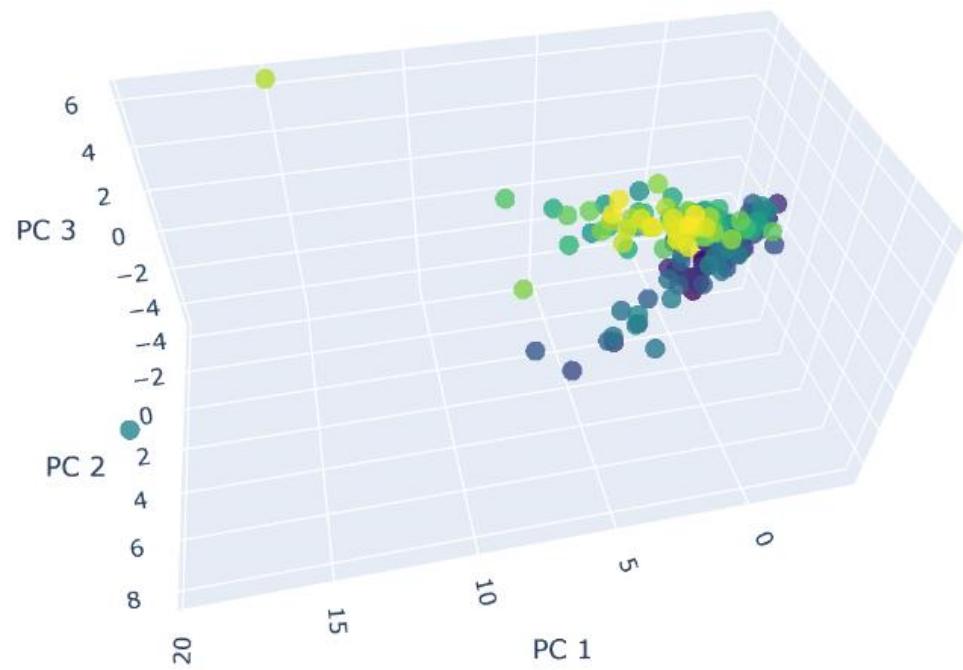
Cells in **dLGN** from **Multiple FOVs** with **48-Stim Set**
PCA, UMAP, Manifold Embedding (reduce **stimuli dim**)

Explained Variance Ratio by Principal Components

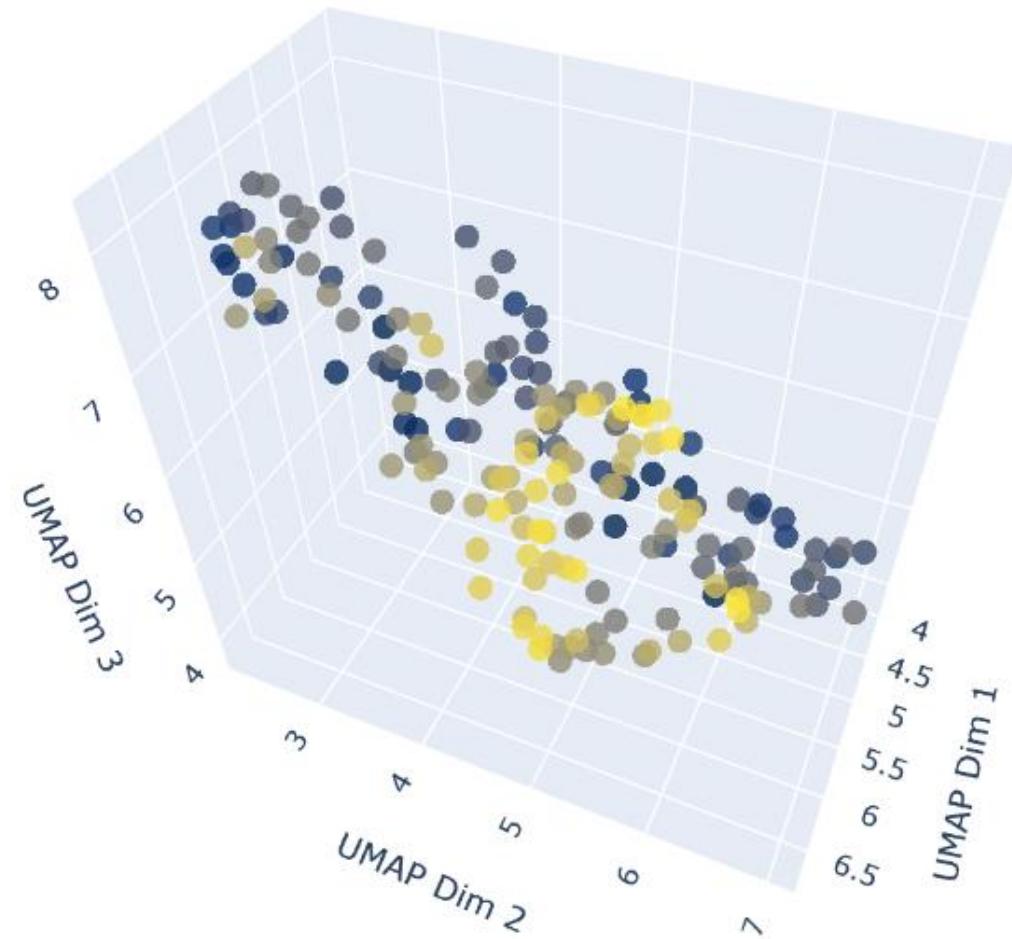


We use the first 3 PCs for plotting.

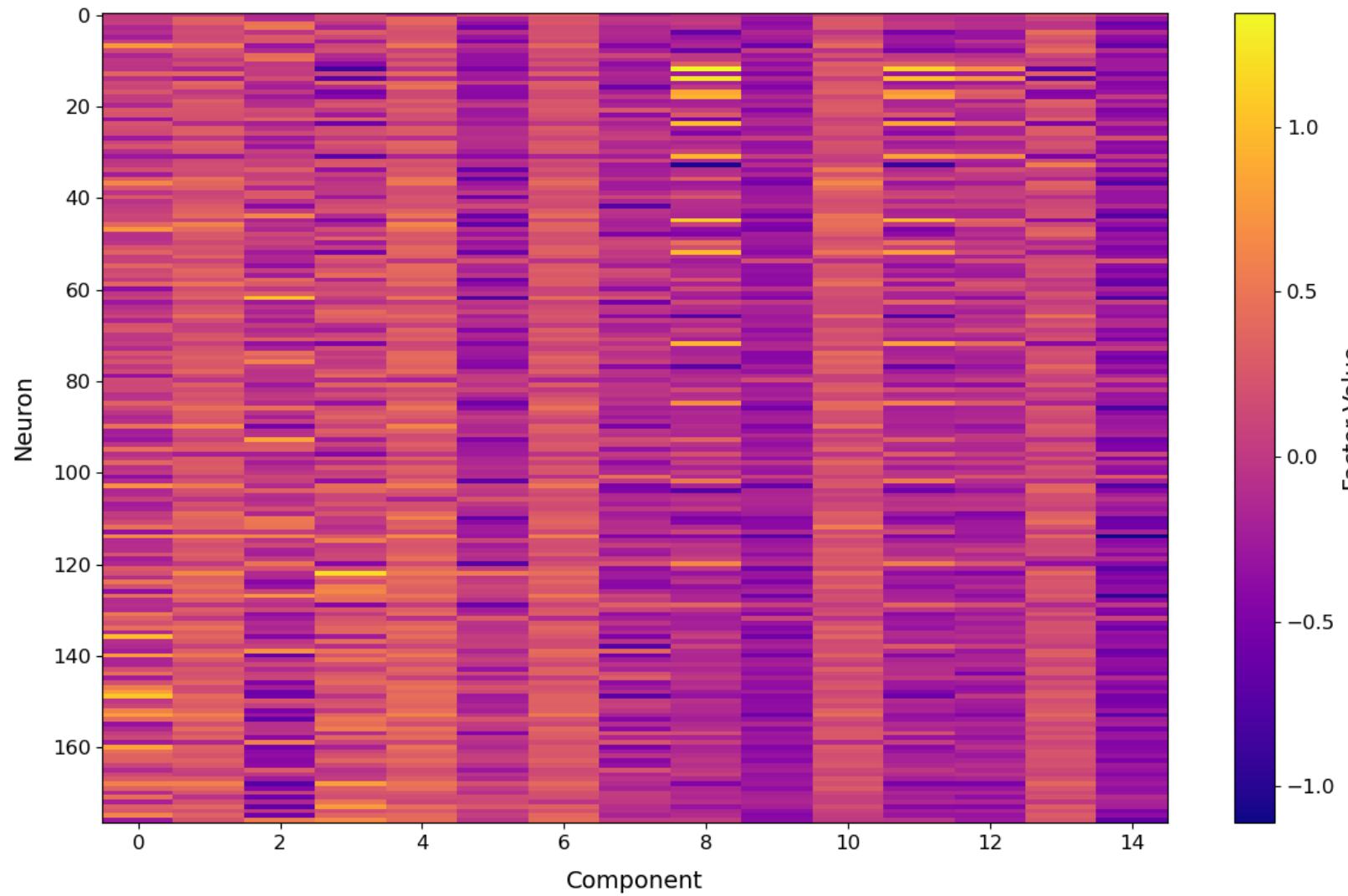
PCA: Neurons in 3D



UMAP: Neurons in 3D



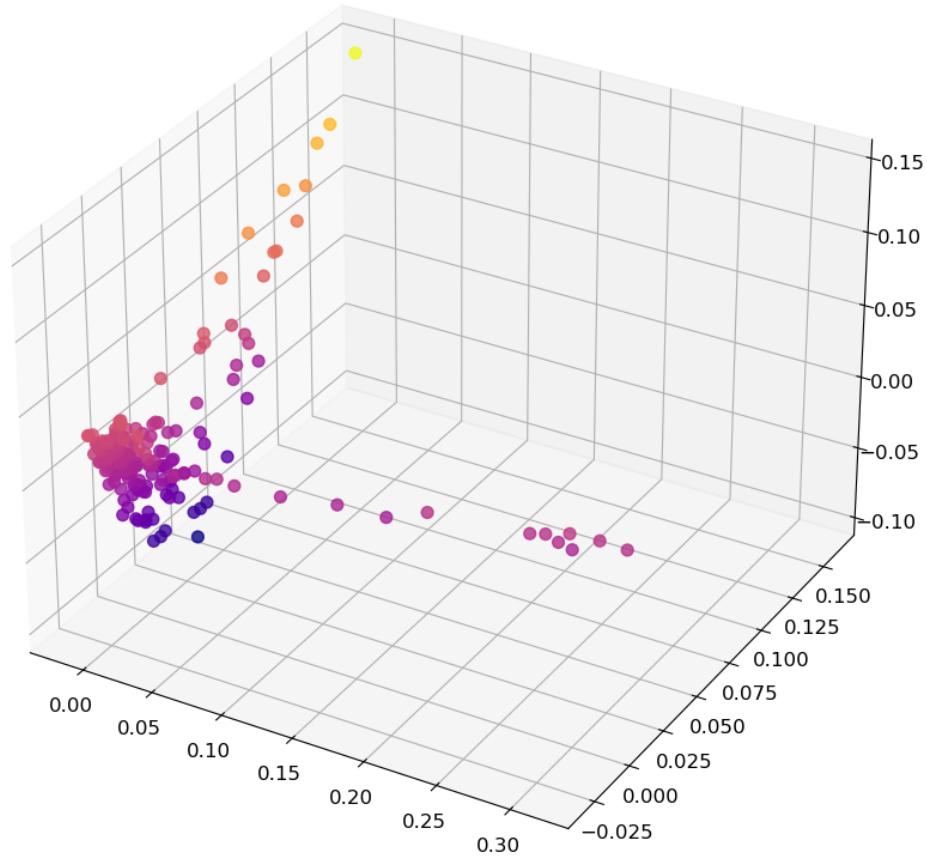
Neuron Factor Matrix



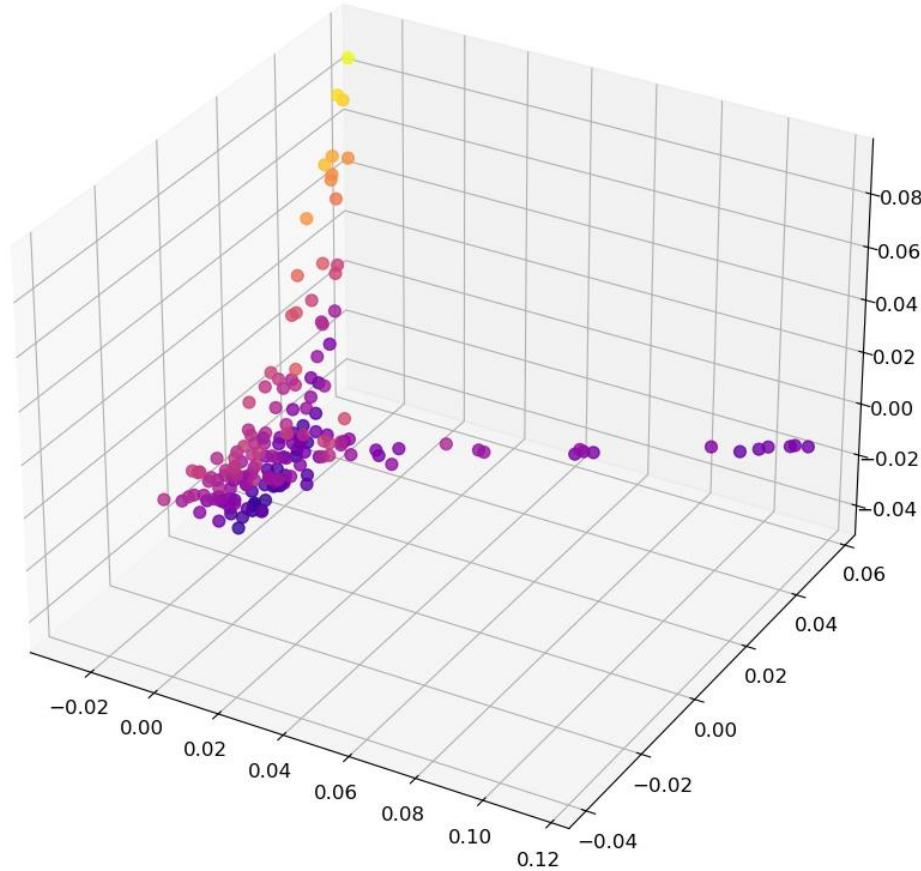
Use the neuron factor matrix to calculate pairwise distance.

Compute Iterated Adaptive Neighborhoods (IAN kernel) from pairwise distances, obtaining data graphs (unweighted and weighted)

Neuron Diffusion Map (Weighted)

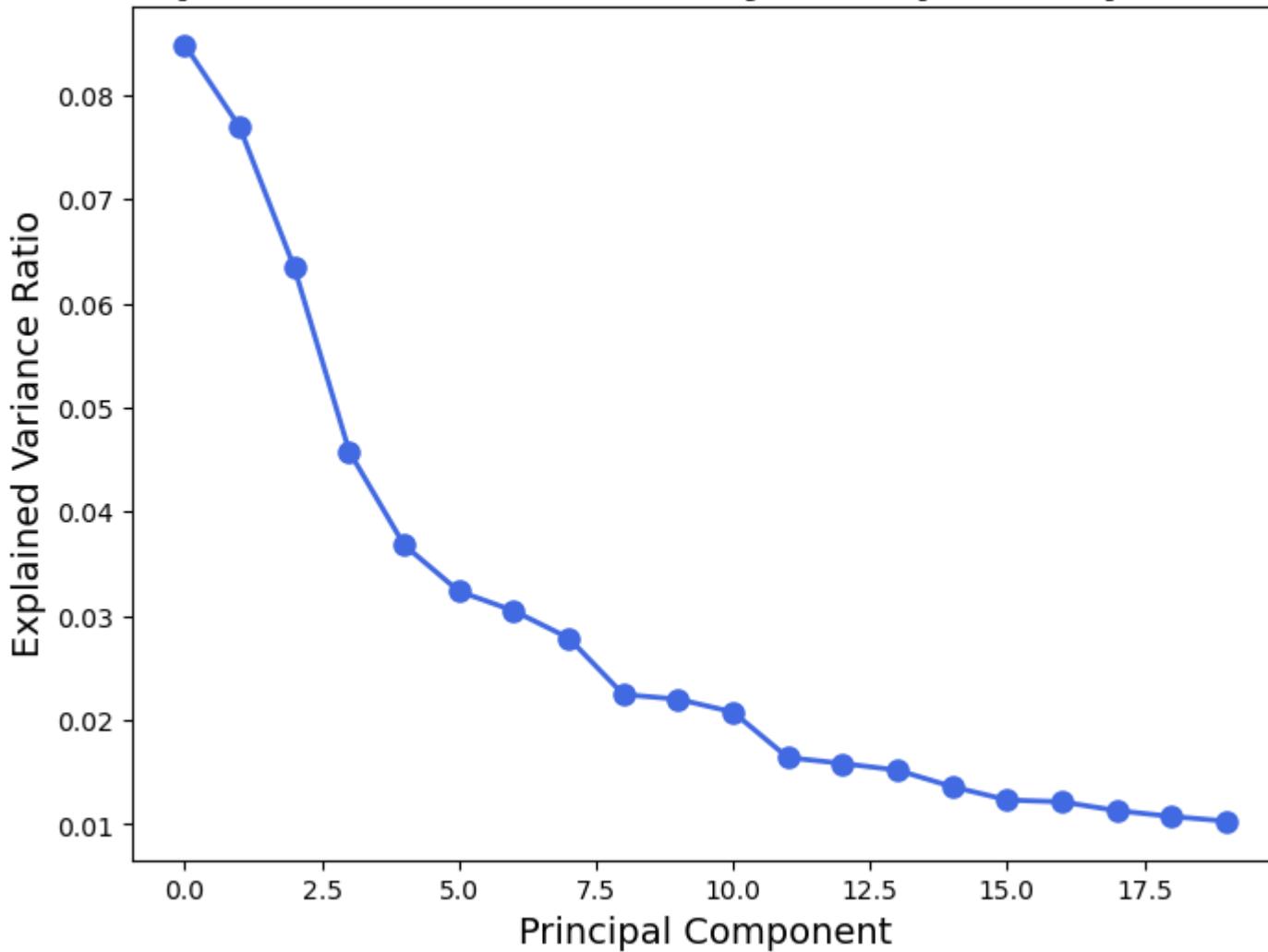


Neuron Diffusion Map (Unweighted)

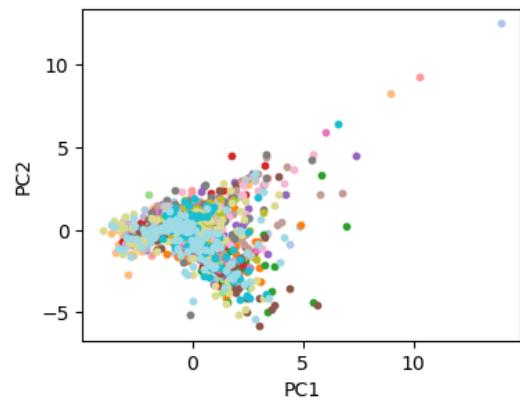
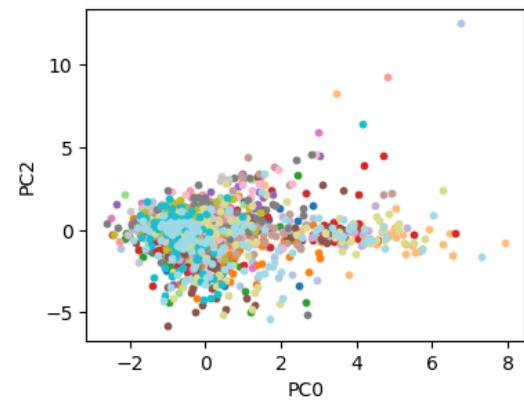
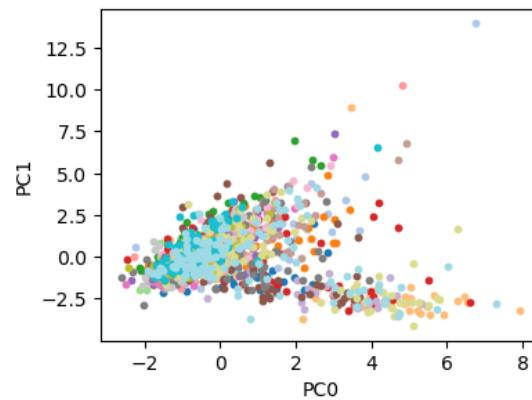


Cells in SC from Multiple FOVs with 48-Stim Set
PCA & UMAP (reduce neurons dim)

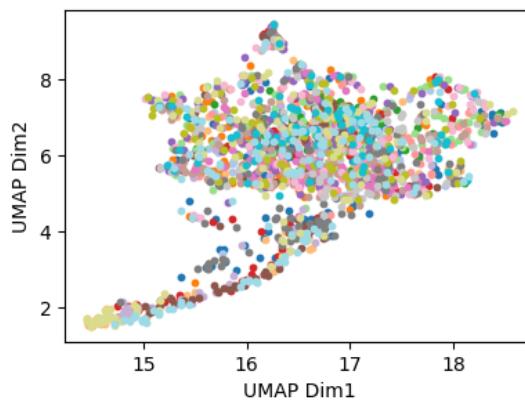
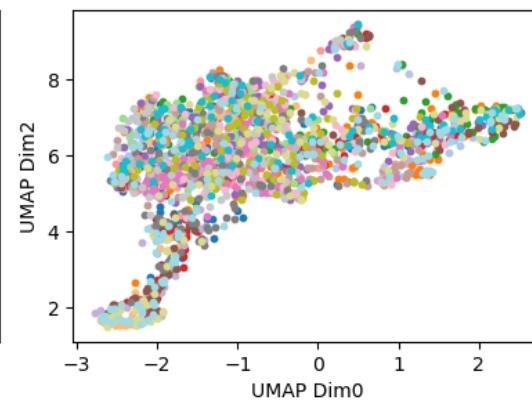
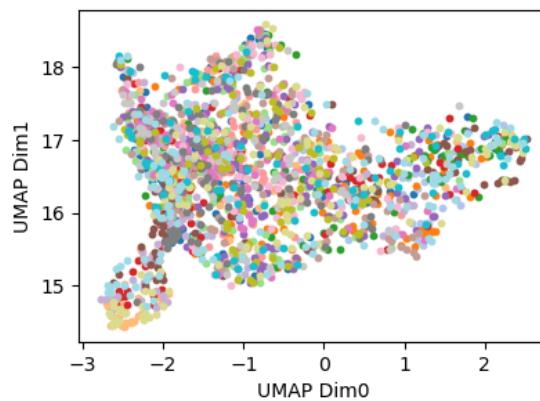
Explained Variance Ratio by Principal Components



PCA: All Stimuli with Labels

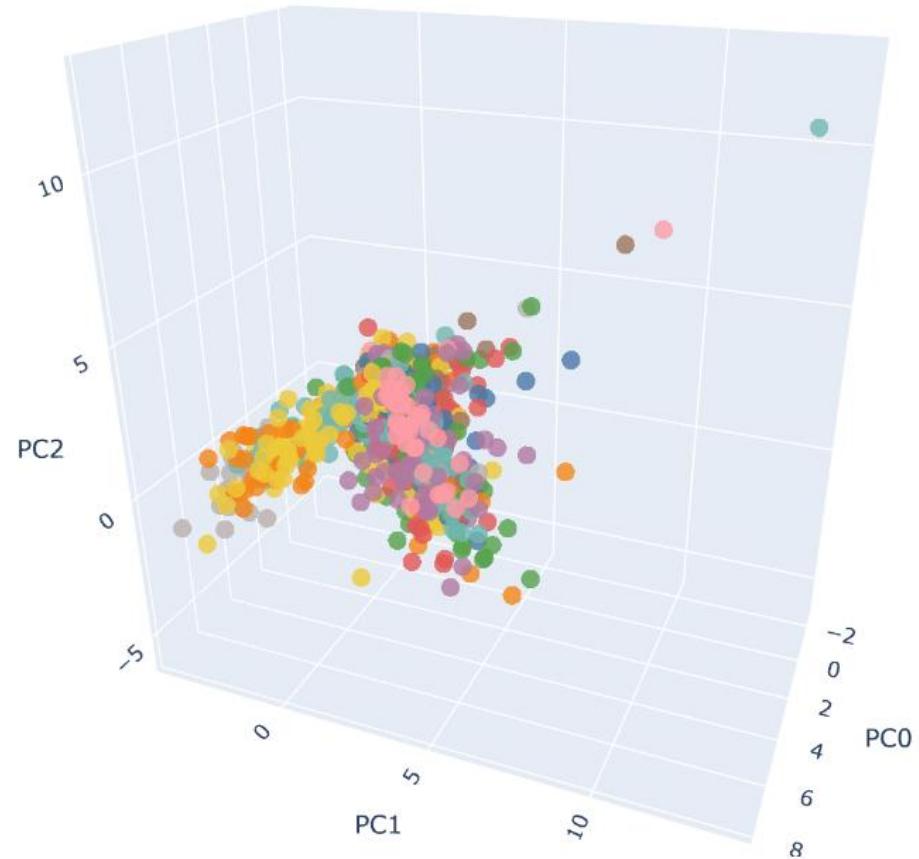


UMAP: All Stimuli with Labels

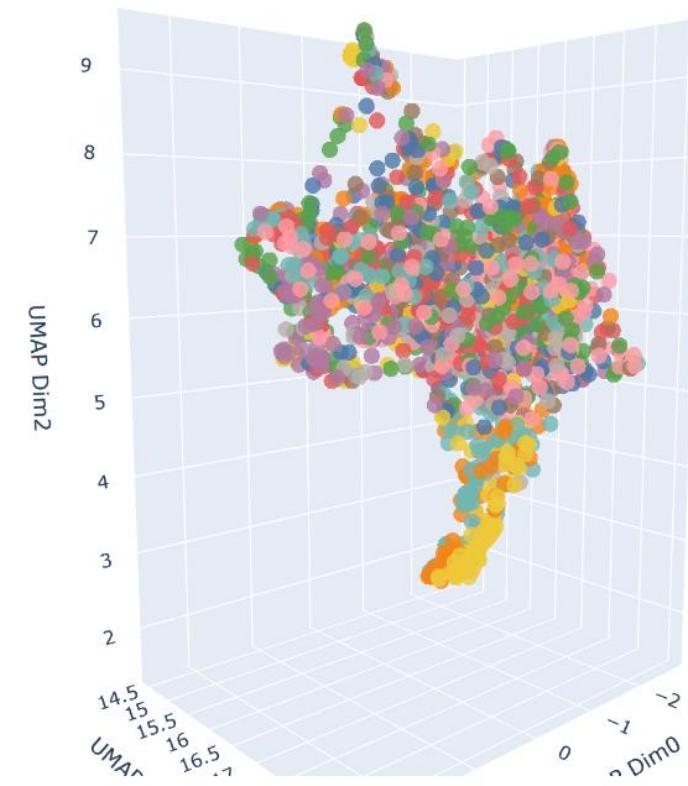


- The figure is a horizontal bar chart with 16 categories on the x-axis. Each category is represented by a colored dot at its start, followed by a horizontal bar indicating the count. The categories are: neg_3flows_45, pos_1flows_45, neg_1flows_0, gratings_hf_0, gratings_lf_180, gratings_hf_225, gratings_lf_270, pos_3flows_270, neg_1flows_90, neg_3flows_315, pos_1flows_0, neg_1flows_135, pos_3flows_225, pos_3flows_225, neg_1flows_180, neg_1flows_225, neg_3flows_225, gratings_hf_315. The counts range from approximately 10 to 150.

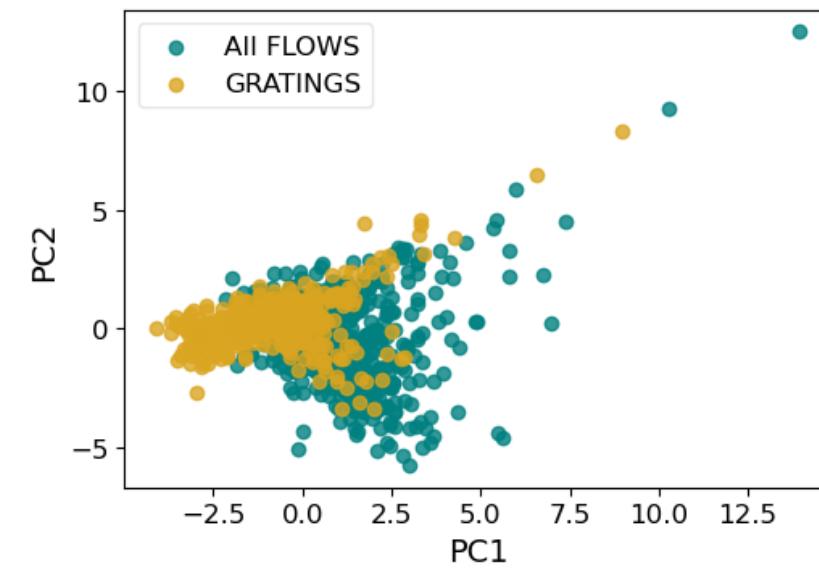
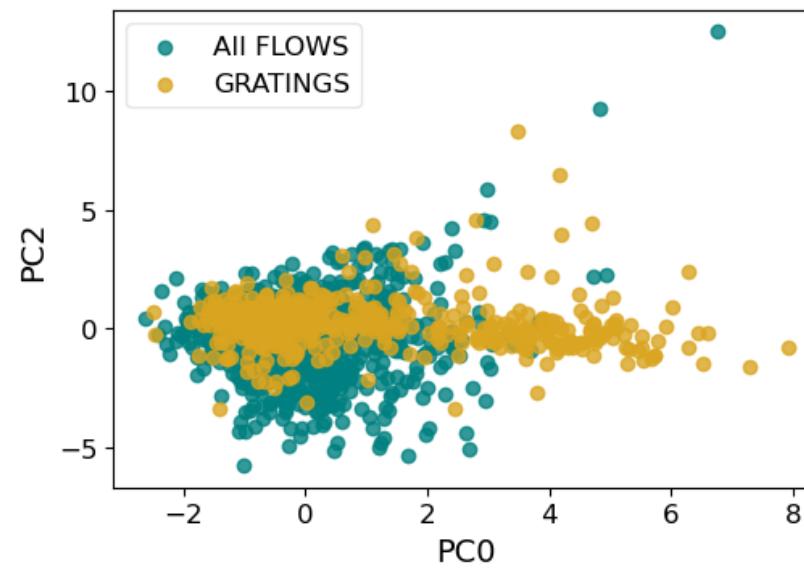
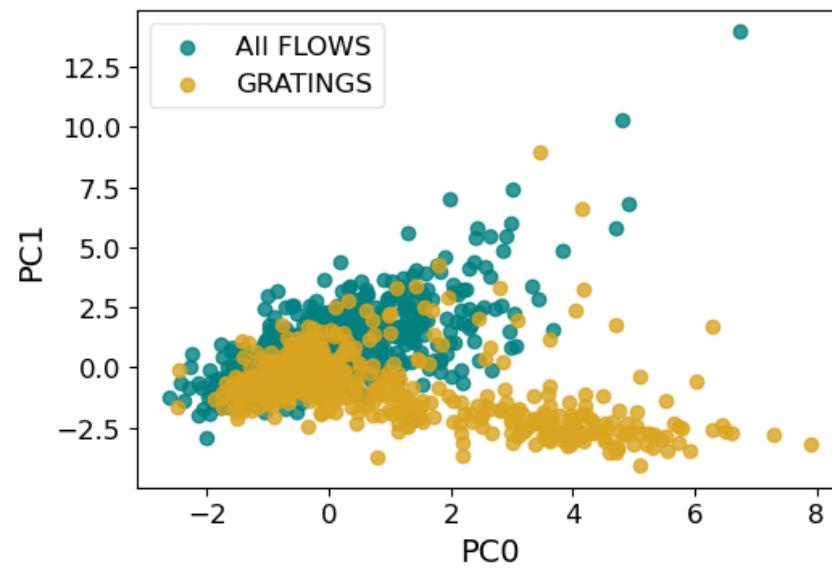
PCA: Interactive 3D Plot of All Stimuli



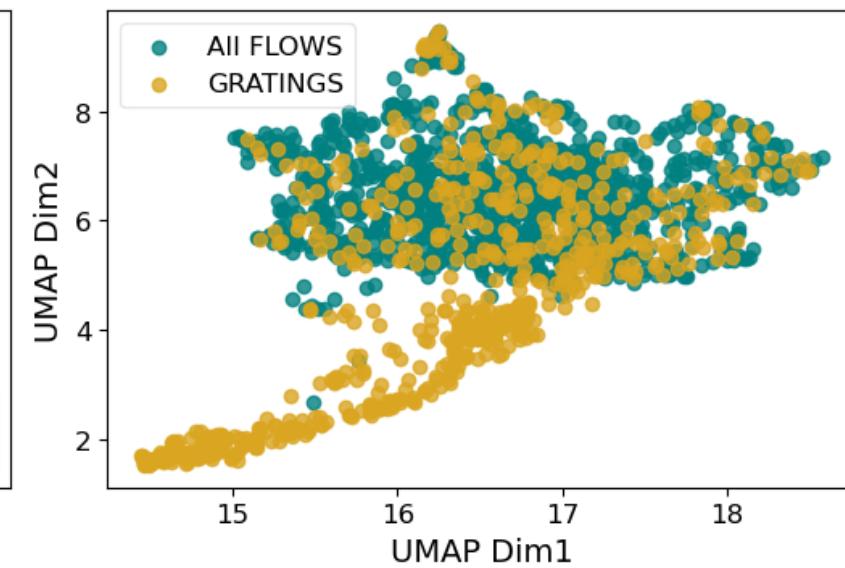
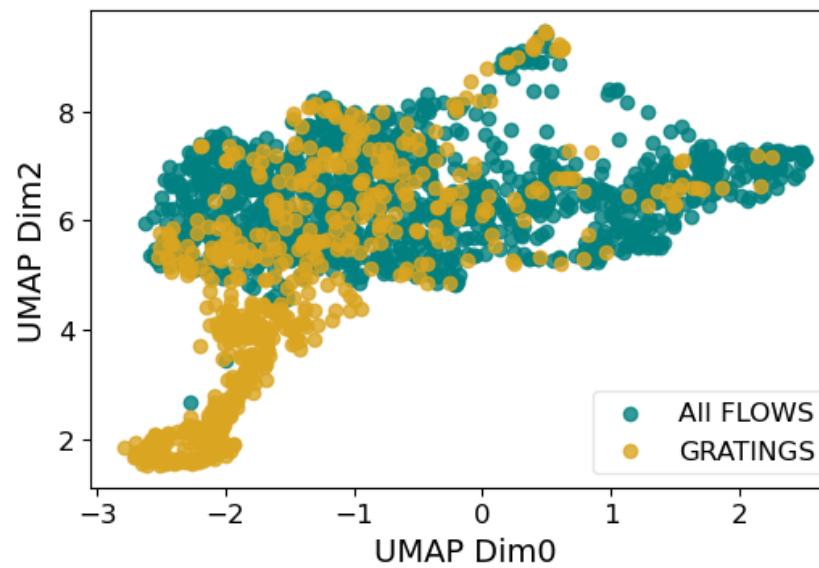
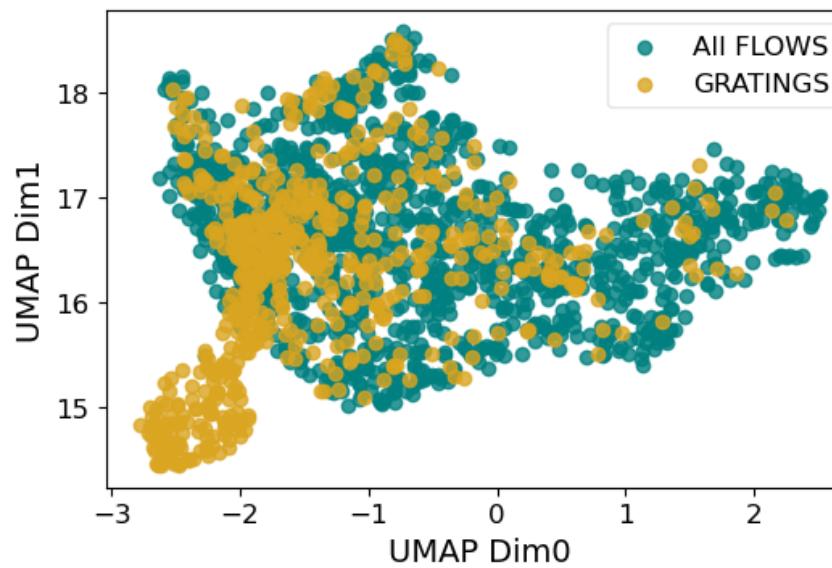
UMAP: Interactive 3D Plot of All Stimuli



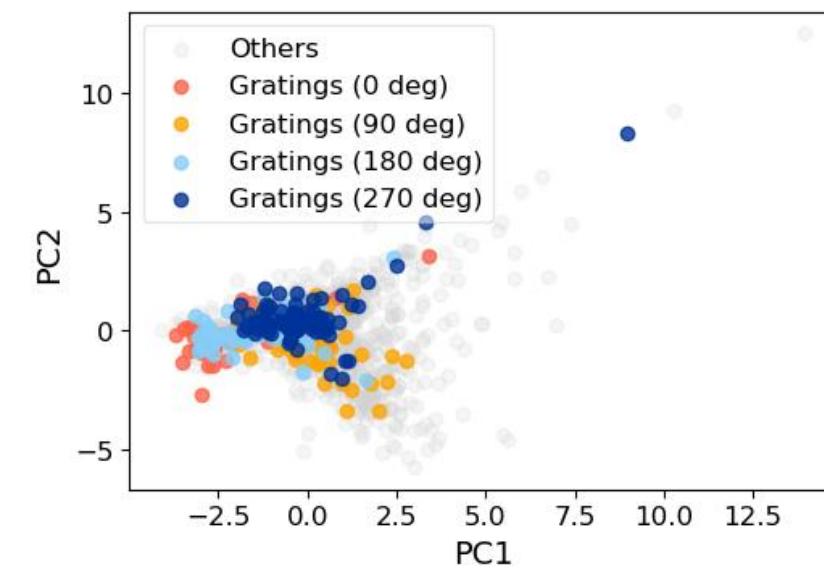
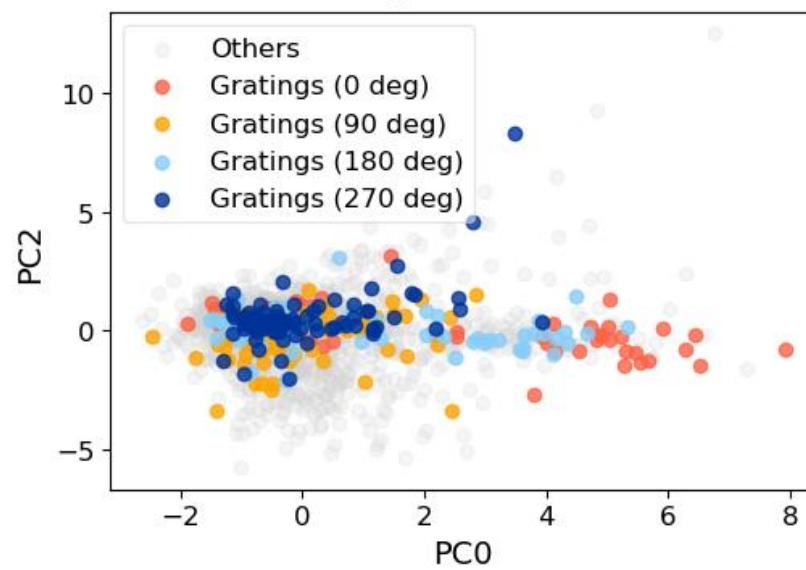
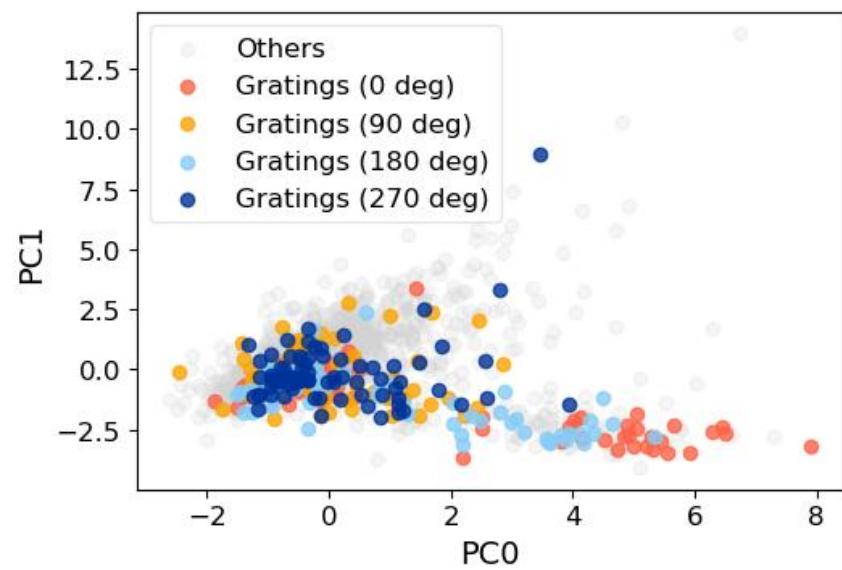
PCA: Gratings vs. All Flows



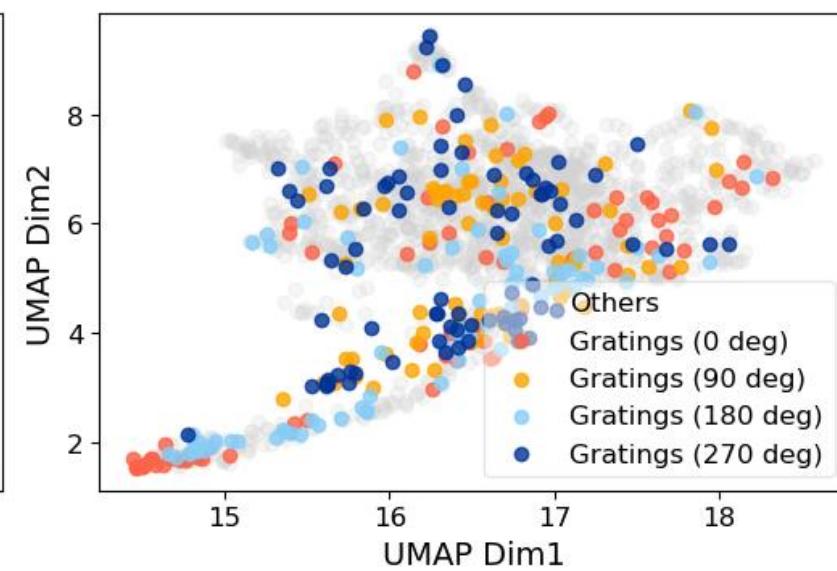
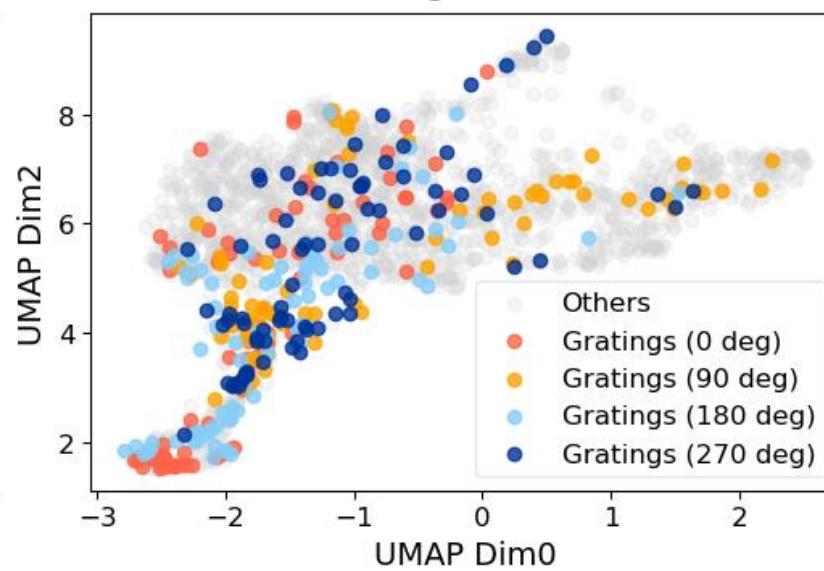
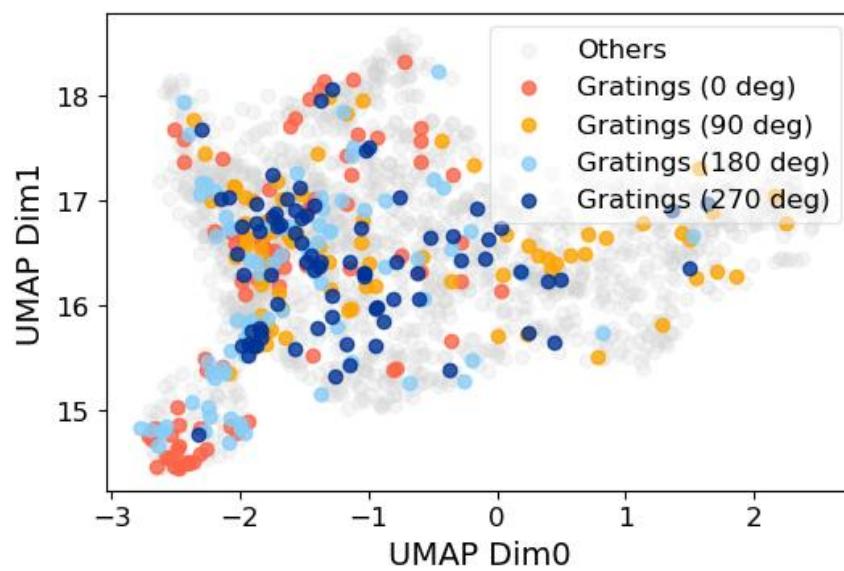
UMAP: Gratings vs. All Flows



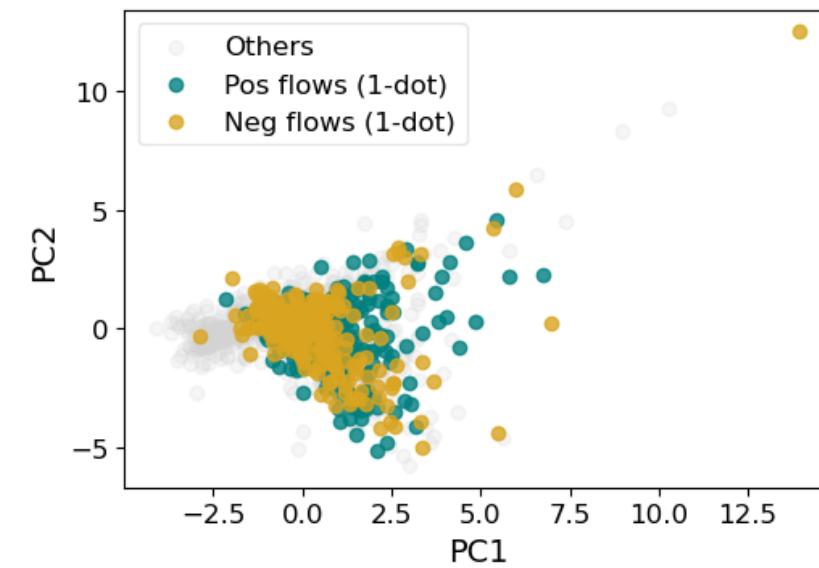
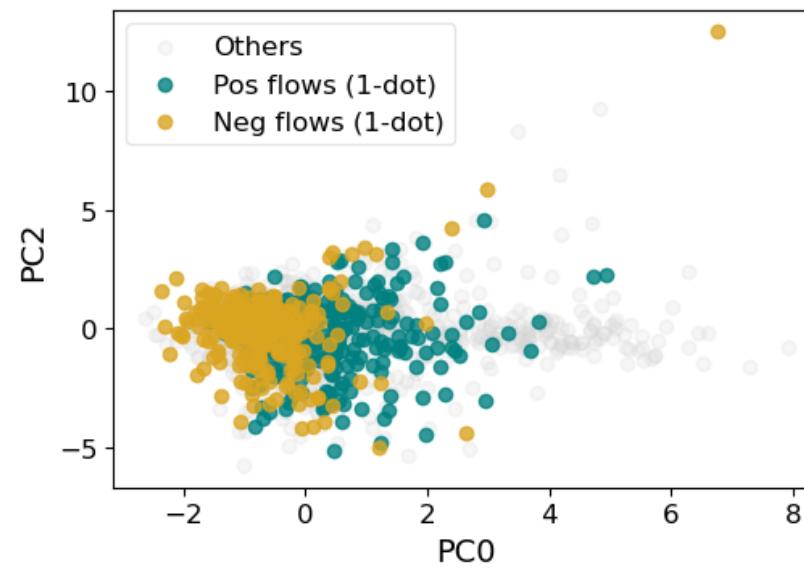
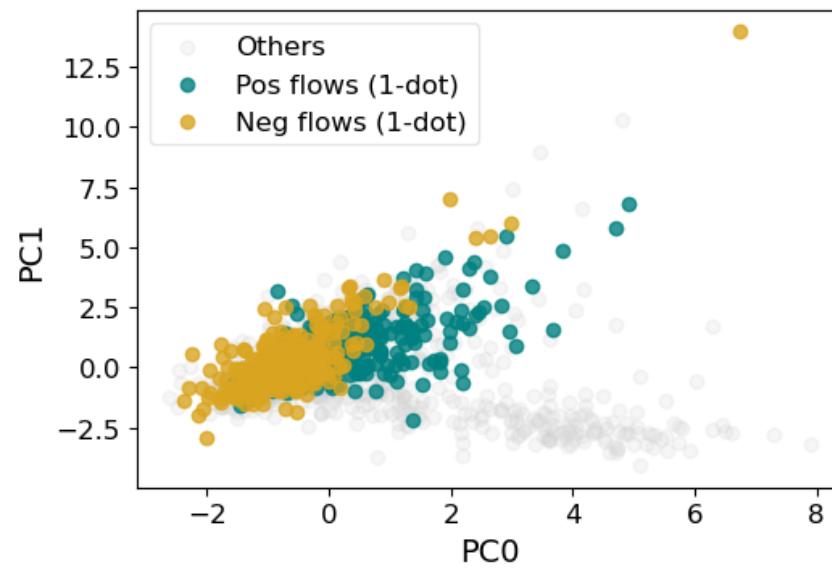
PCA: Gratings directions



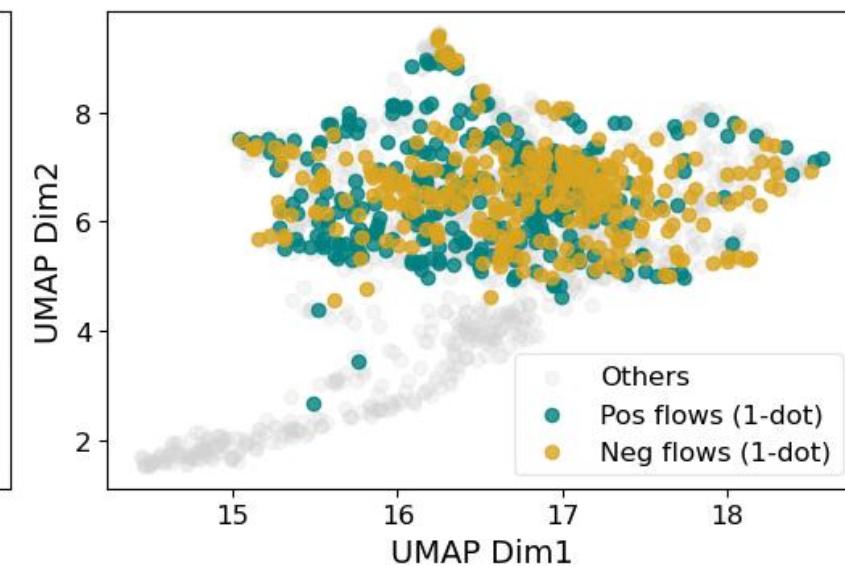
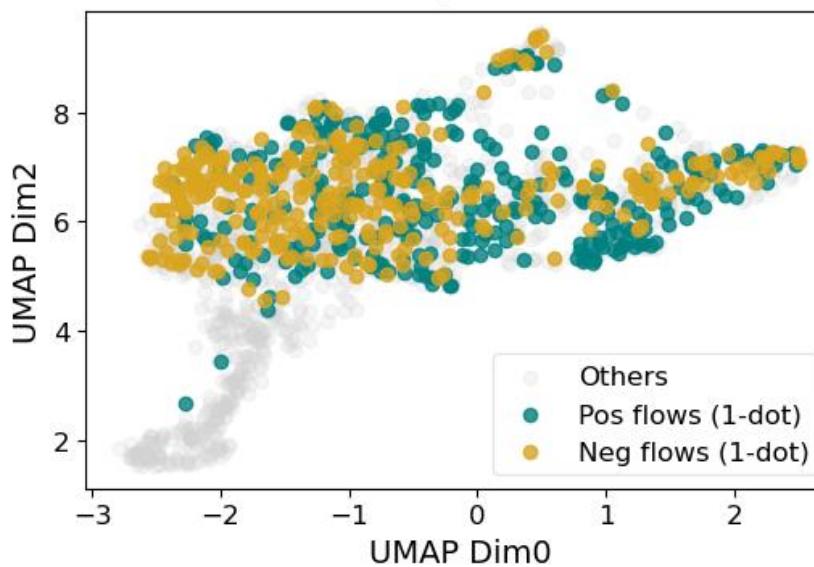
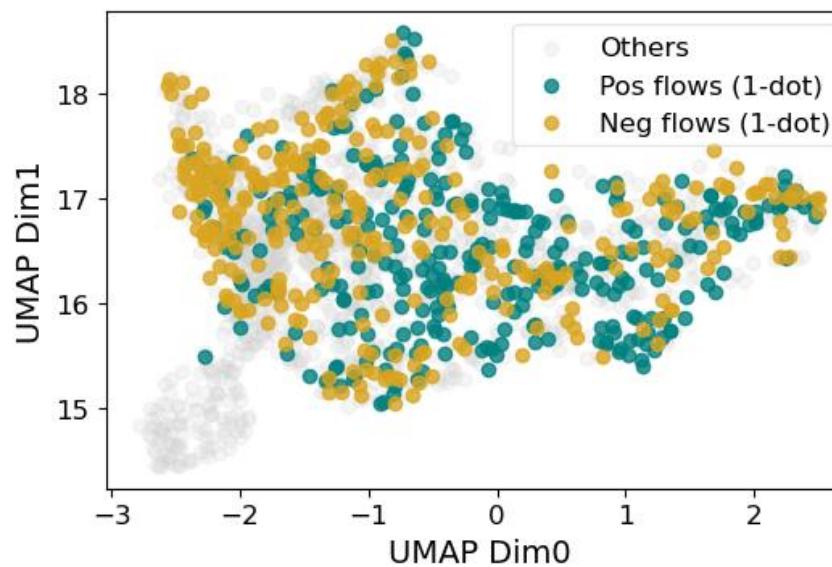
UMAP: Gratings directions



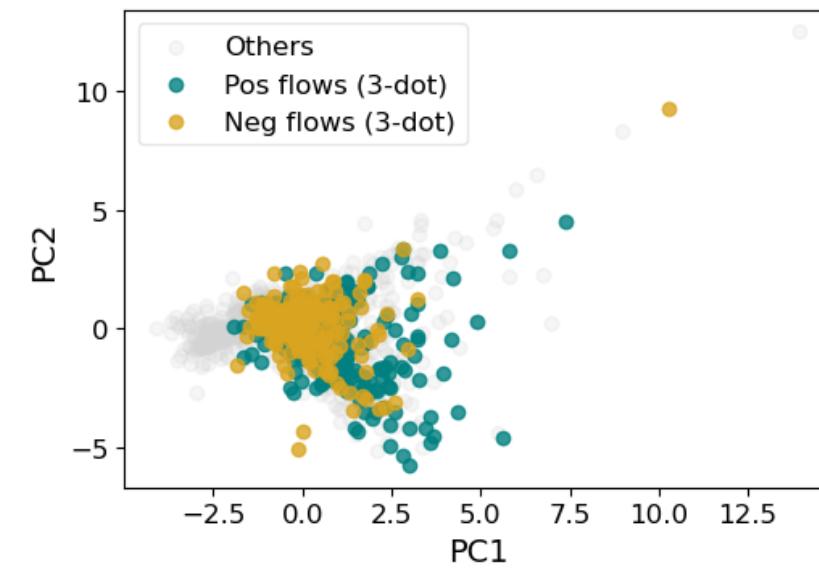
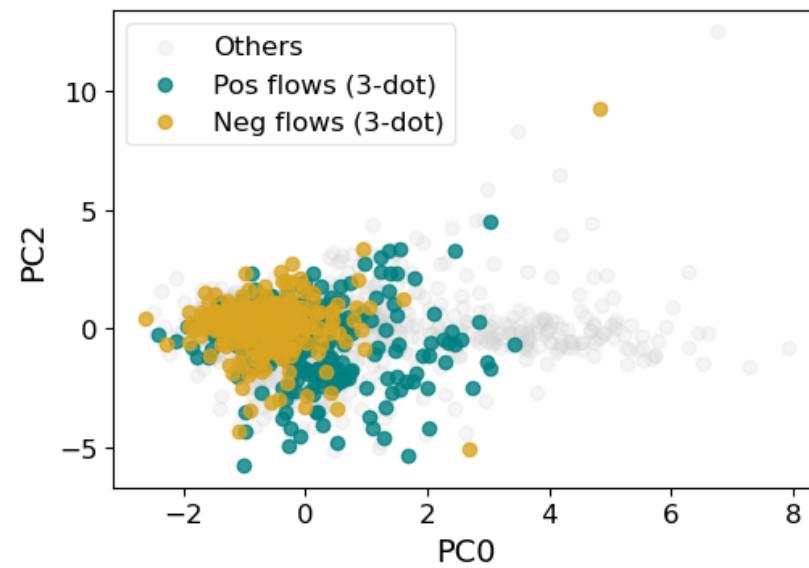
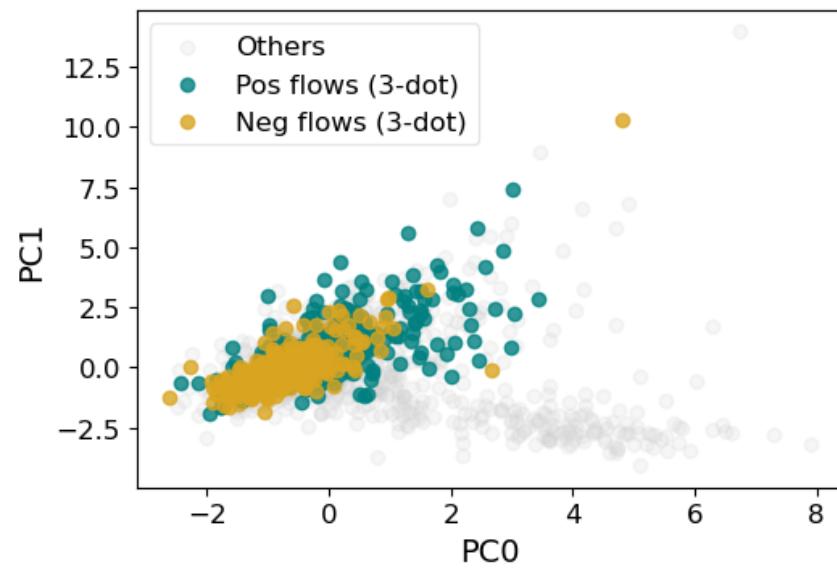
PCA: Pos vs neg flows (1-dot)



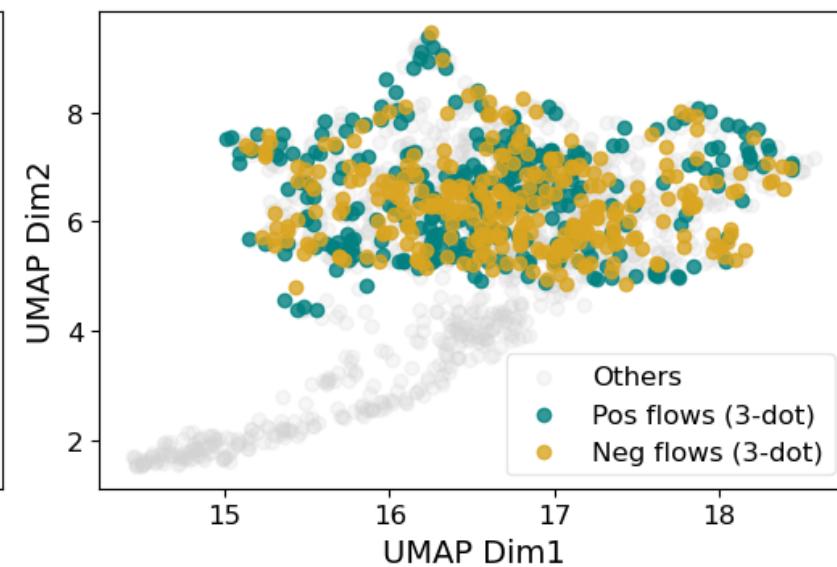
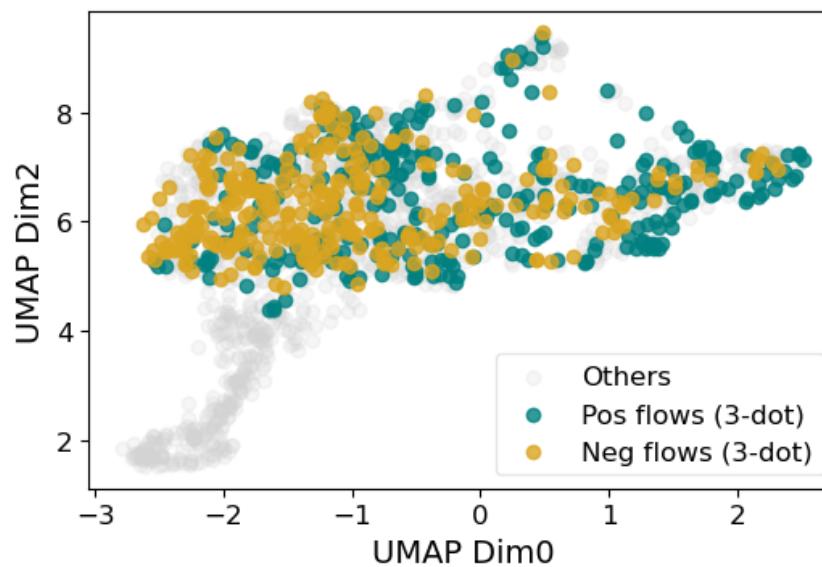
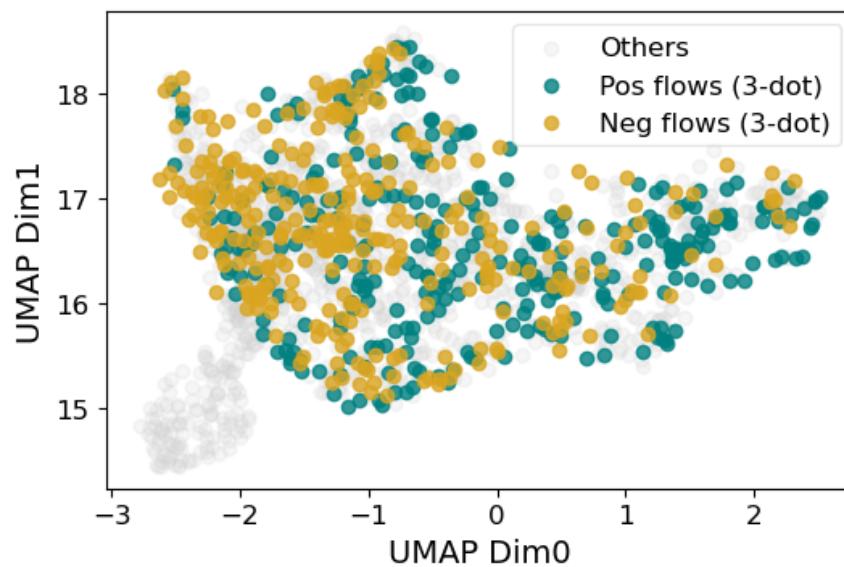
UMAP: Pos vs neg flows (1-dot)



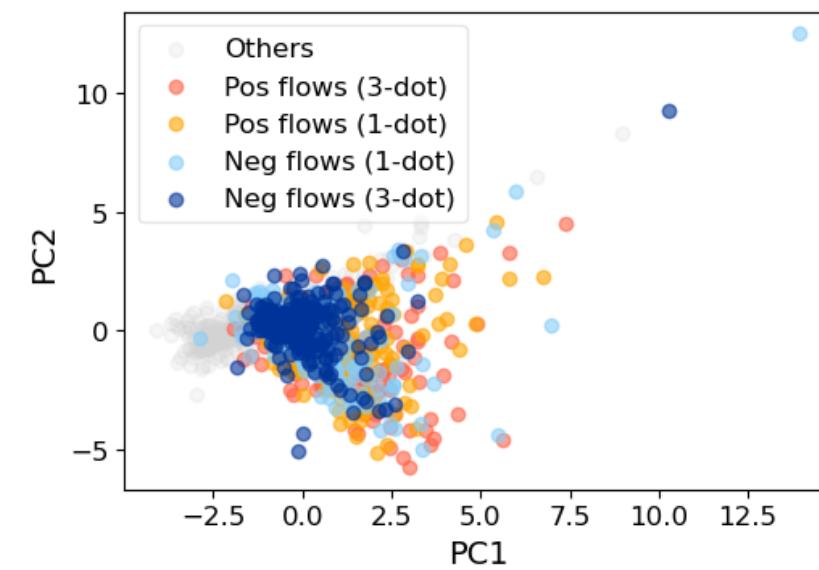
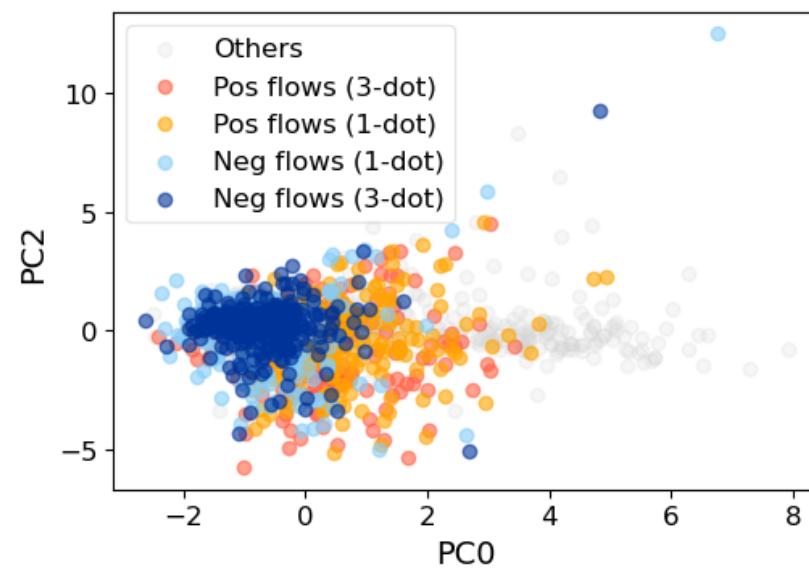
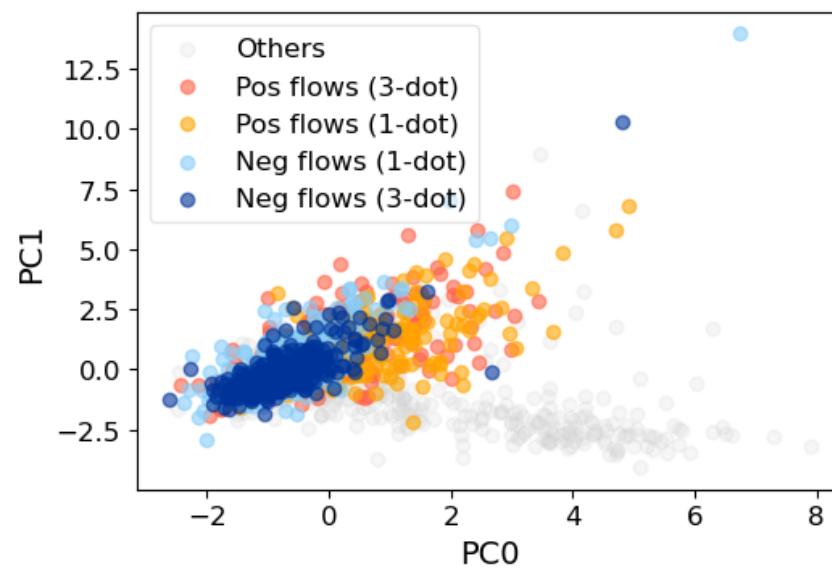
PCA: Pos vs neg flows (3-dot)



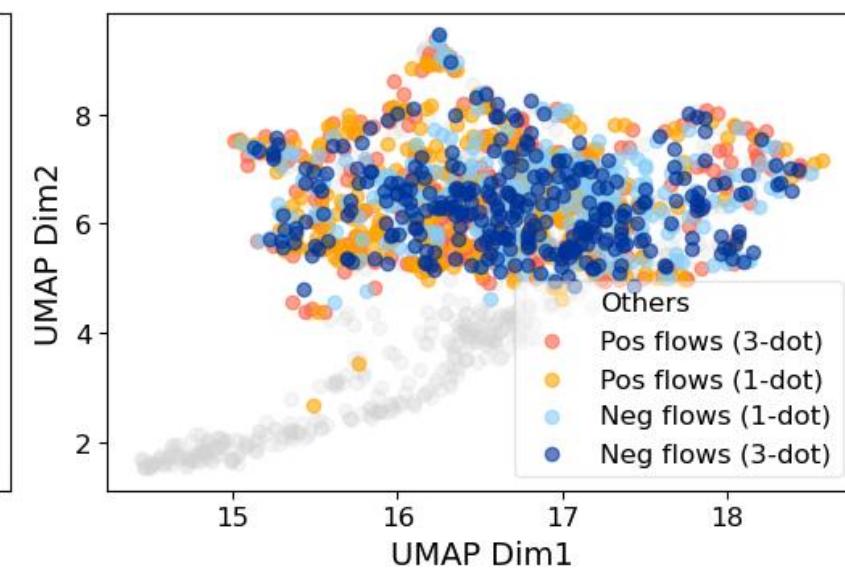
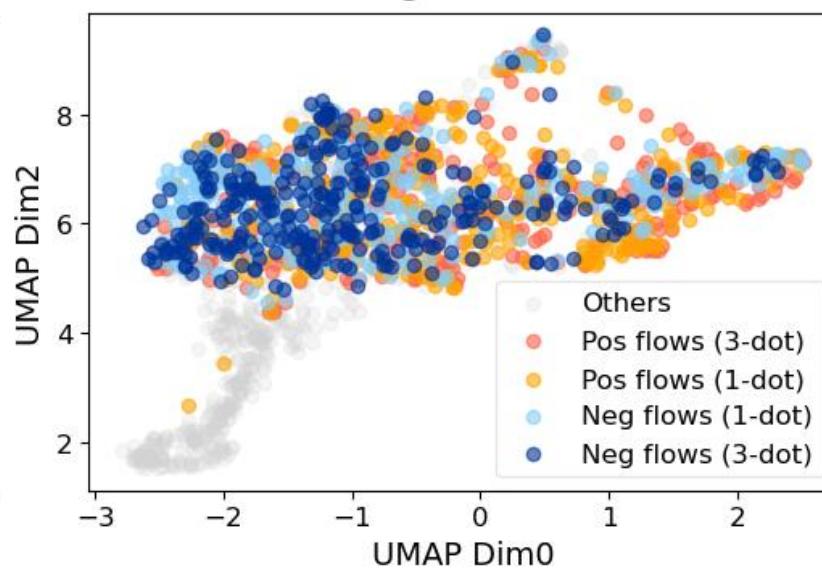
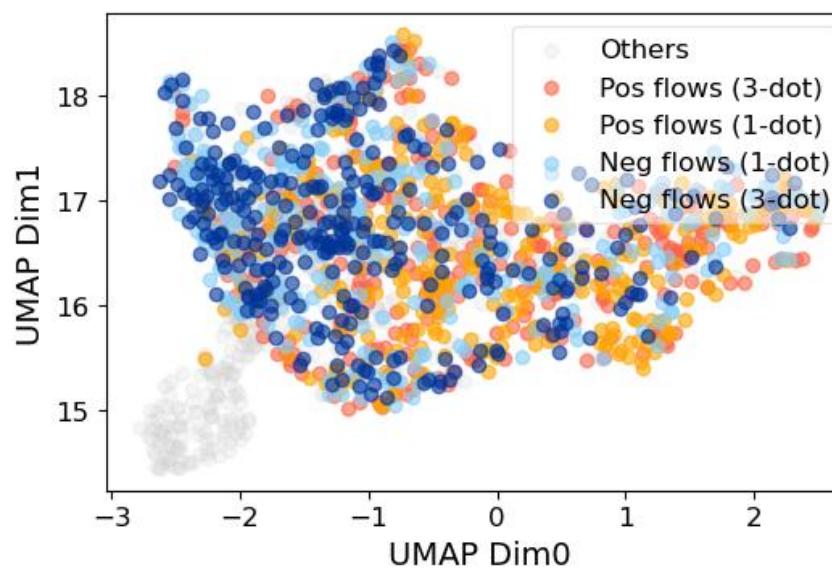
UMAP: Pos vs neg flows (3-dot)



PCA: Pos vs neg flows (1 & 3-dot)

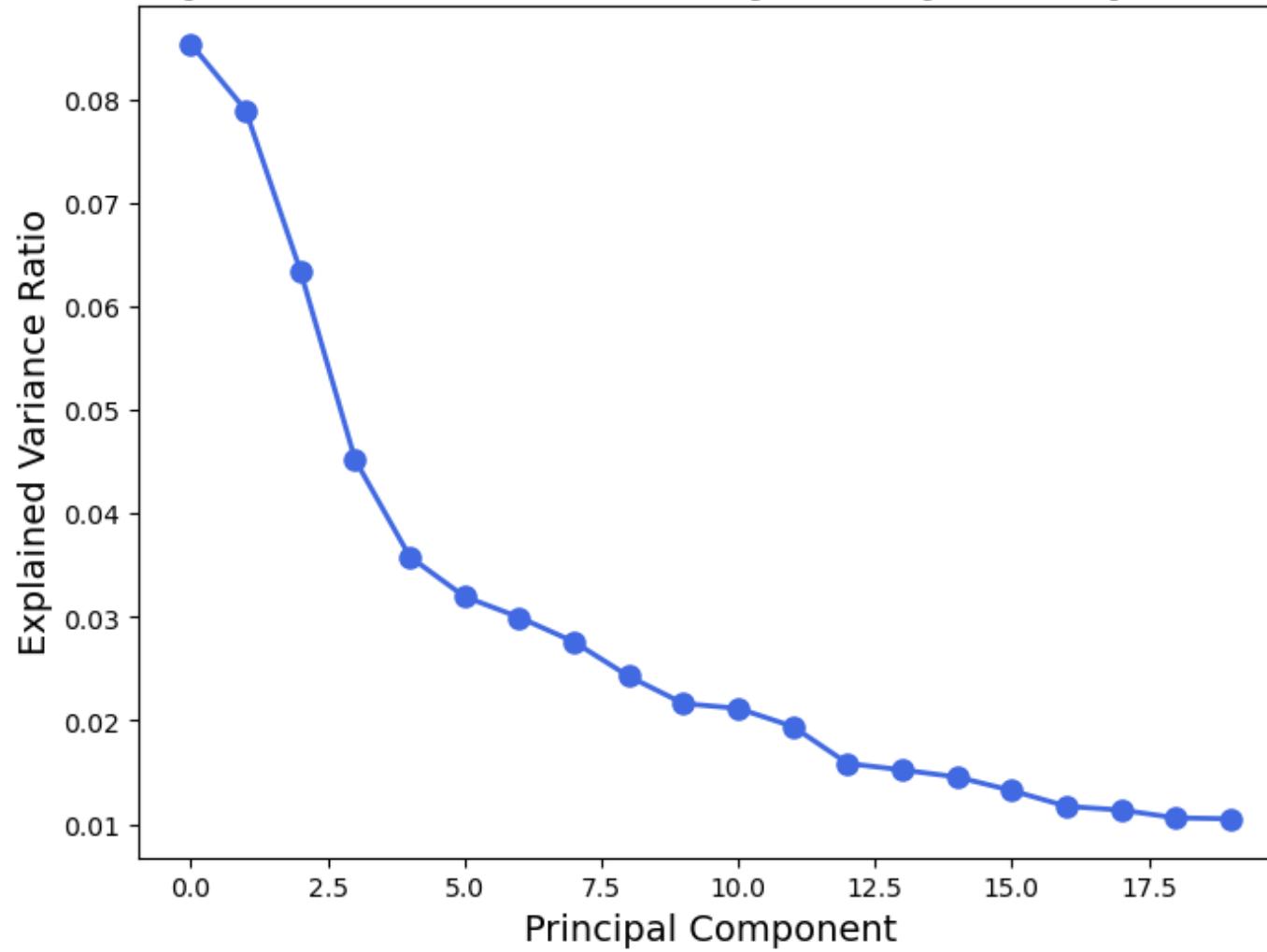


UMAP: Pos vs neg flows (1 & 3-dot)

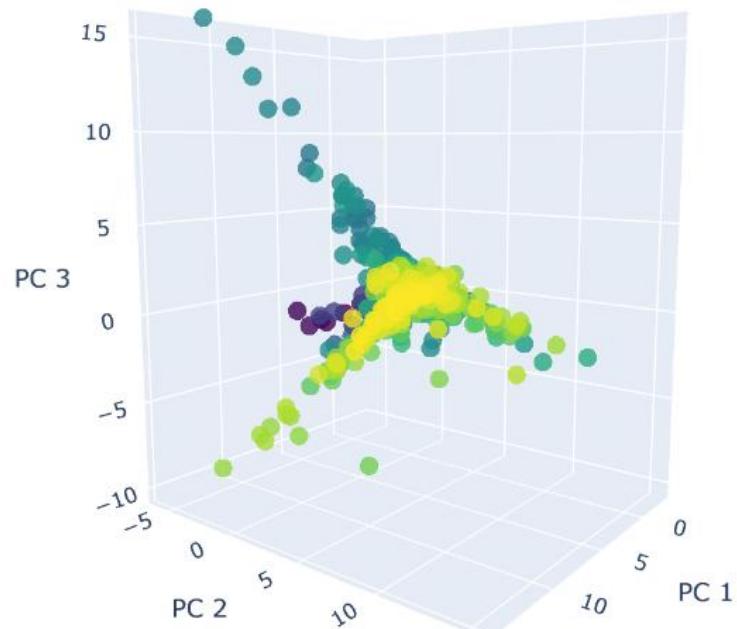


Cells in **SC** from **Multiple FOVs** with **48-Stim Set**
PCA, UMAP, Manifold Embedding (reduce **stimuli dim**)

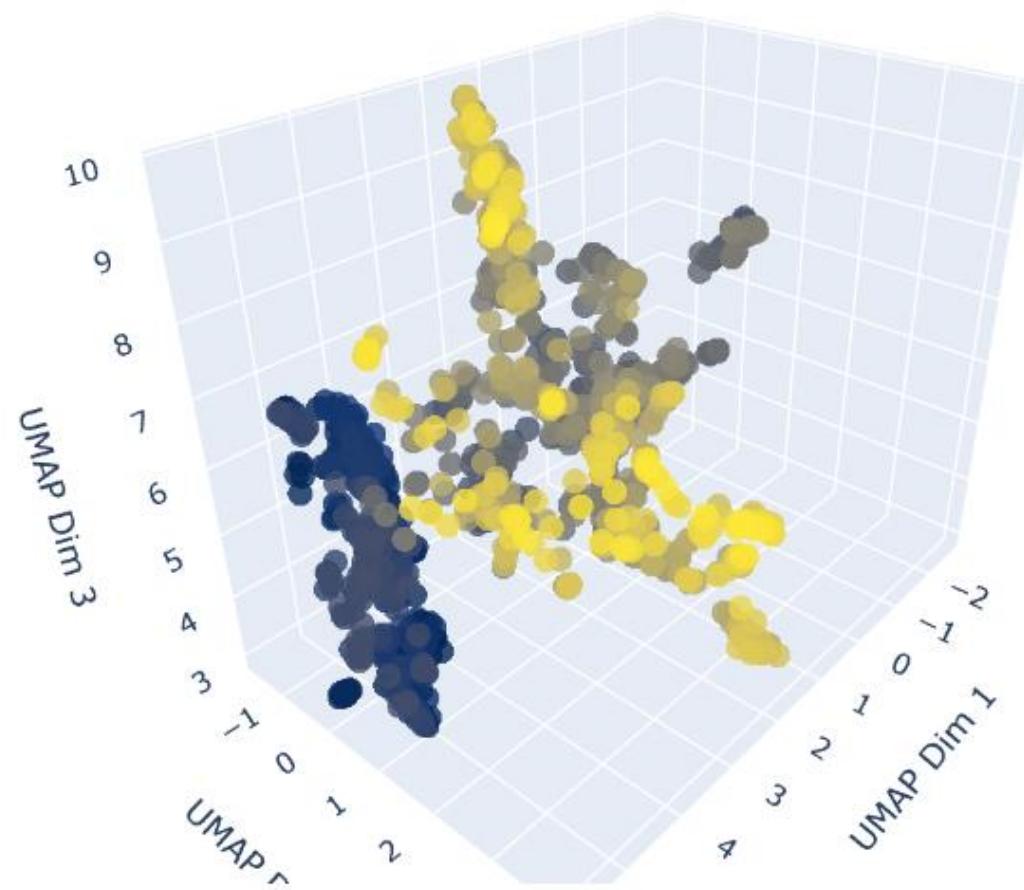
Explained Variance Ratio by Principal Components



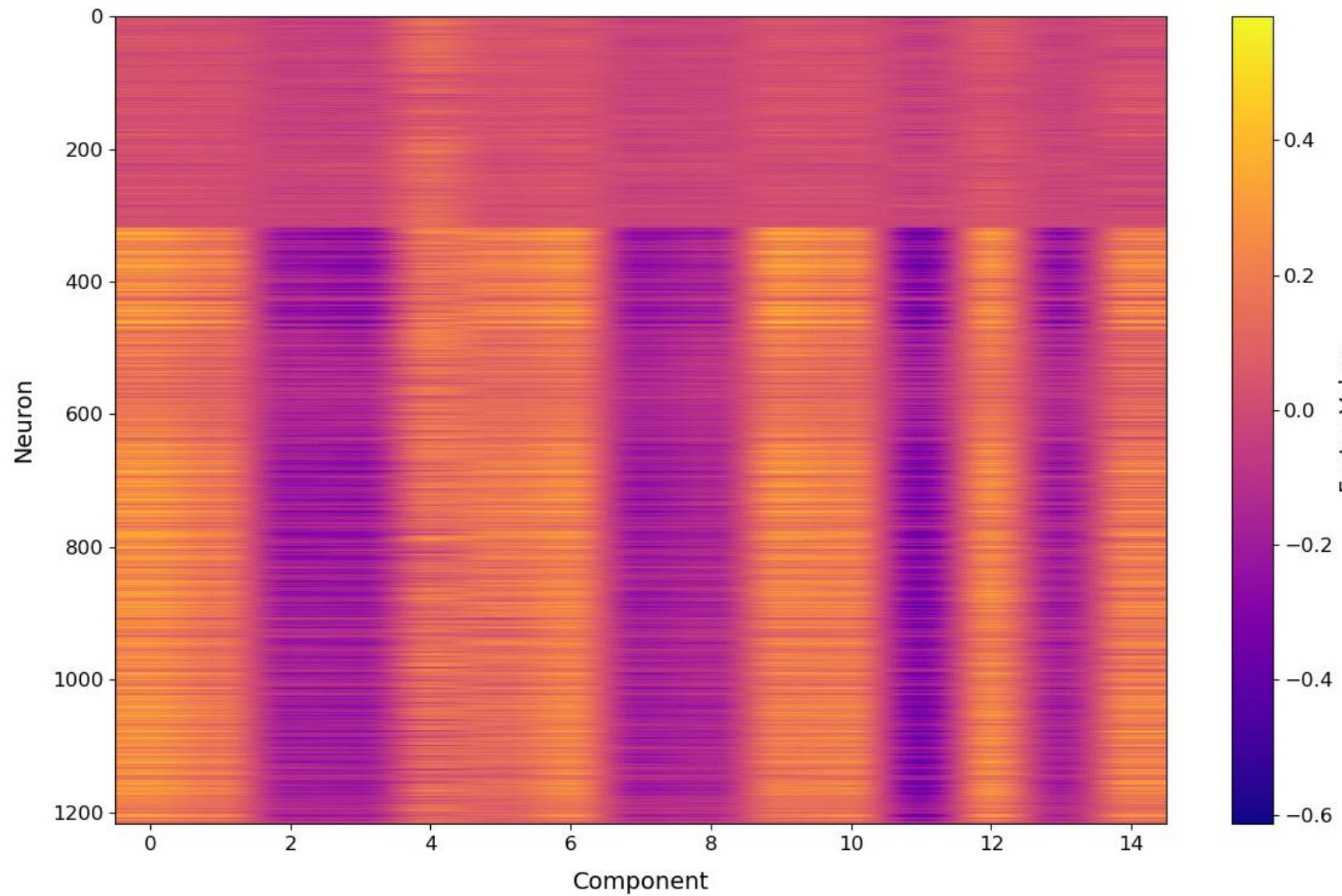
PCA: Neurons in 3D



UMAP: Neurons in 3D



Neuron Factor Matrix

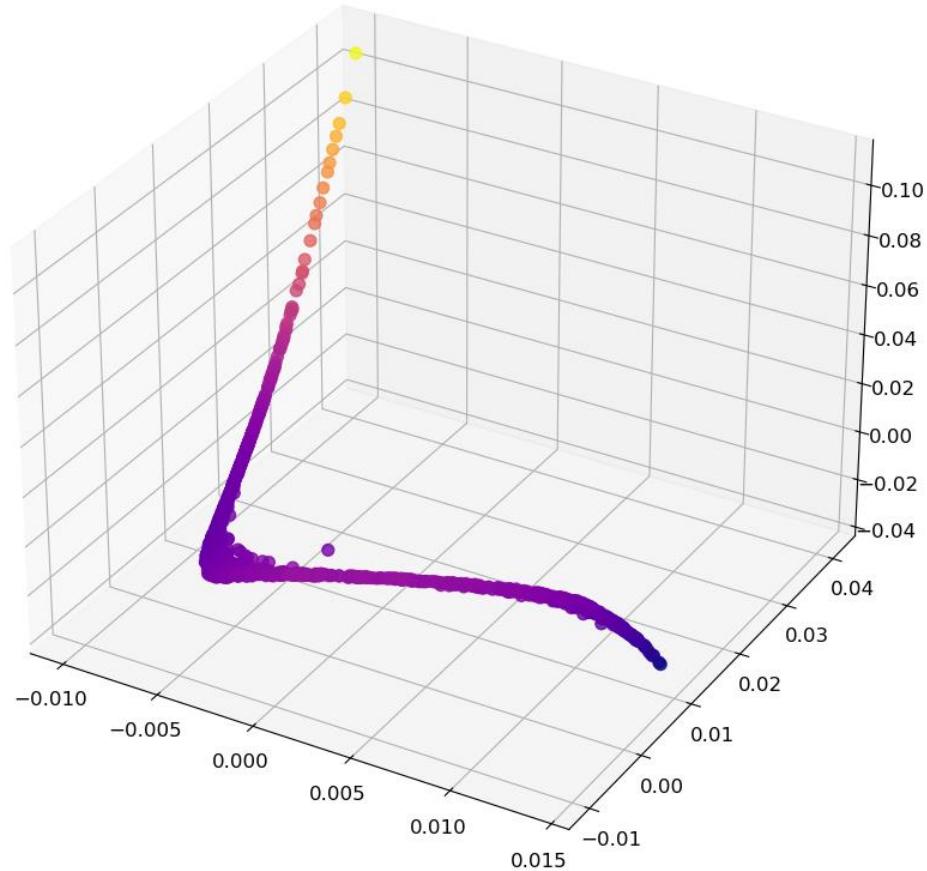


Two types of neurons

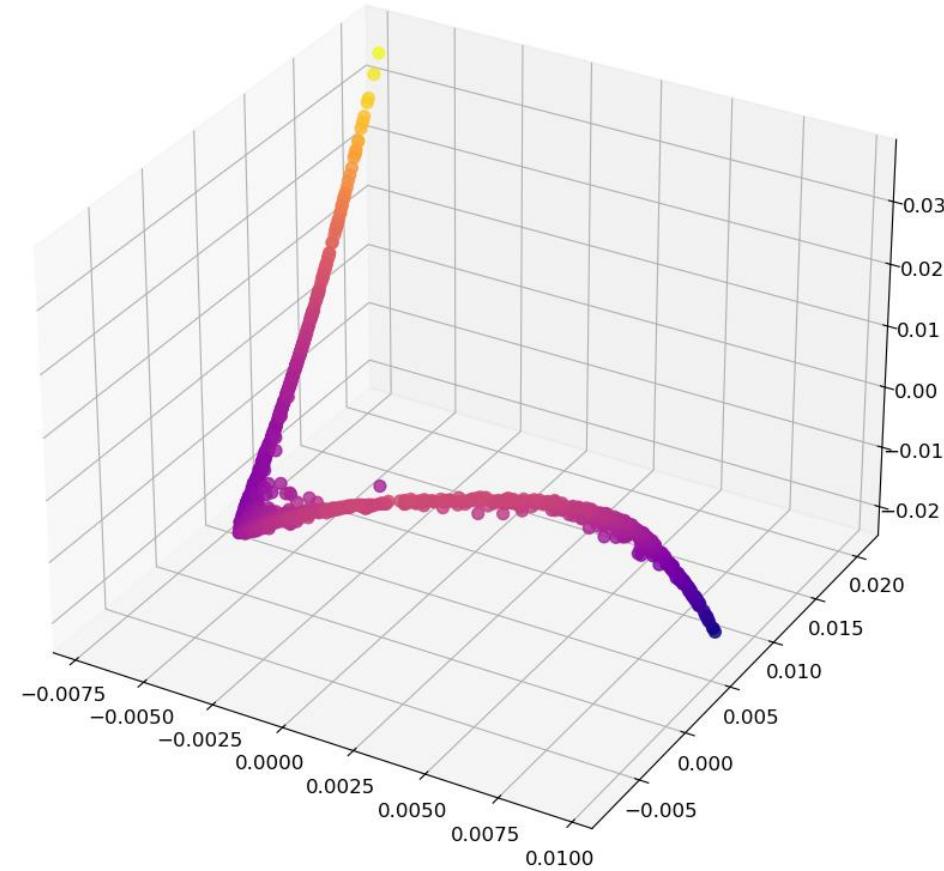
Use the neuron factor matrix
to calculate pairwise
distance.

Compute Iterated Adaptive
Neighborhoods (IAN kernel)
from pairwise distances,
obtaining data graphs
(unweighted and weighted)

Neuron Diffusion Map (Weighted)



Neuron Diffusion Map (Unweighted)



Thanks