In this notebook, we will simulate the dissimilarity measured from distance in neural response space generated from sigmoid tuning of multi-neurons encoding.

```
import numpy as np
import matplotlib.pyplot as plt
```

Tuning neurons

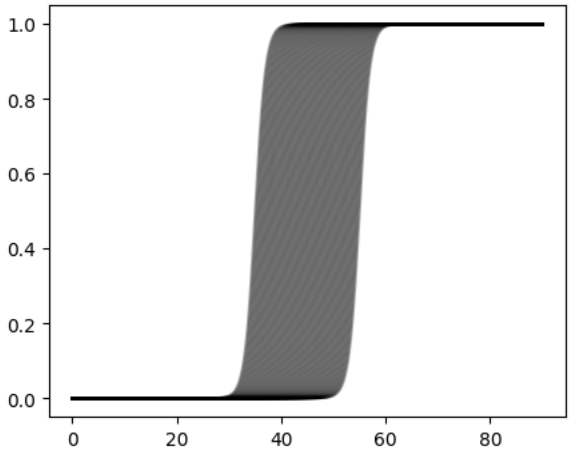
```
def logit(x: float | np.ndarray, x0: float = 0, lower: float = 0, upper: float = 1, slope: float = 1) -> f
    return np.power(1+np.exp(-(x-x0)), -1)
```

```
NUM_NEURONS: int = 91
# ACTIVATION_VALS: np.ndarray = np.linspace(0,90,NUM_NEURONS)
ACTIVATION_VALS: np.ndarray = np.linspace(35,55, NUM_NEURONS)
NUM_STIM_VALS: int = 1001
STIMULUS_VALUES: np.ndarray = np.linspace(0,90,NUM_STIM_VALS)
```

```
ACTIVATION_VALS[10]
```

```
37.22222222222222
```

```
fig, ax = plt.subplots(figsize=(5,4))
for i in range(NUM_NEURONS):
    current_mid_value = ACTIVATION_VALS[i]
    tuning_curve = logit(STIMULUS_VALUES, current_mid_value)
    ax.plot(STIMULUS_VALUES, tuning_curve, alpha = 0.3, c="black", label=f"i")
plt.show()
```



Representational space

```
response_list = []
for i in range(NUM_NEURONS):
    current_mid_value = ACTIVATION_VALS[i]
    tuning_curve = logit(STIMULUS_VALUES, current_mid_value)
    response_list.append(tuning_curve)
response_space = np.vstack(response_list).T
```

```
dist_list = []
for i in range(NUM_STIM_VALS):
    diff = response_space[i] - response_space[0]
    dist = np.linalg.norm(diff)
    dist_list.append(dist)
```

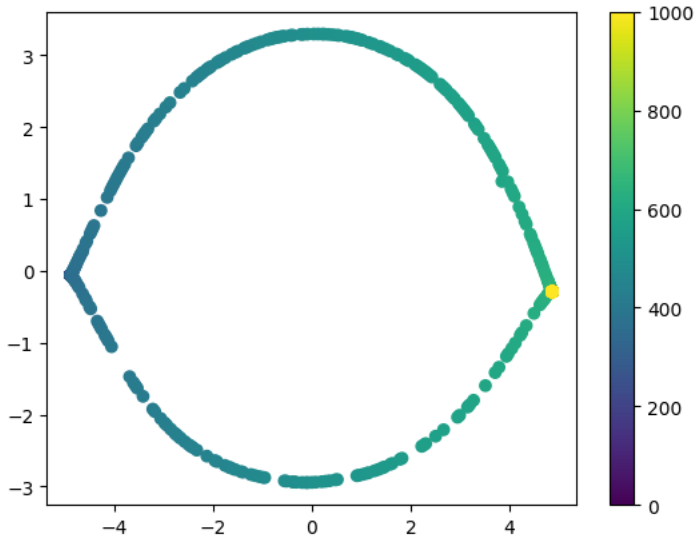
MDS

```
from sklearn.manifold import MDS
```

```
embedding = MDS(n_components=2, normalized_stress='auto')
response_space_transformed = embedding.fit_transform(response_space)
```

```
plt.scatter(response_space_transformed[:,0], response_space_transformed[:,1], c = np.arange(NUM_STIM_VALS))
plt.colorbar()
```

```
<matplotlib.colorbar.Colorbar at 0x7f4515ae16f0>
```




```
1
```