CISC 3250
Systems Neuroscience

Representations in the brain

Professor Daniel Leeds
dleeds@fordham.edu
JMH 332

How do we represent our world?
Diverse sensations

Dog
• Body parts
  – tail, ears, legs
• Sounds
  – bark, whimper, pant
• Feel
  – fur

Flower
• Appearance
  – color, size, shape
• Smell
• Feel
  – texture, temperature

We call each piece of information a “feature”

How do we represent our world?
One sensation, multiple levels

Song
• Meaning of words
• Pitch/melody
• Rhythm
• Language
• Singer identity

Dance
• Body part
  – arms, hands, legs
• Direction
  – forward, to-the-left
• Timing
  – order of moves, speed

Computational representations describing a visual object

• A picture is worth a million pixels
  – Digital picture broken into a grid of boxes – pixels
  – Each pixel contains a color

• Translate from pixels to category label:
  floss flour flower flume flute foam
Data in the brain

- Neural location related to information encoded
- Progression of encoding for increasingly complex concepts

Simple outline of vision pathway

1. Retina: pixel detectors
2. Primary visual cortex (V1): edge detectors
3. Second-cortical layer (V2?): edge combination detectors
   ...   
N. Higher-cortical layer: Full-object detectors

Interacting representations: feedforward network

- More-complex information/features computed from simpler information/features

Edge detector in action

\[ w_1 = -0.5 \]
\[ w_2 = -1 \]
\[ w_3 = -0.5 \]
\[ w_4 = 0.5 \]
\[ w_5 = 1 \]
\[ w_6 = 0.5 \]
\[ w_7 = -0.5 \]
\[ w_8 = -1 \]
\[ w_9 = -0.5 \]
Cortical organization and feature organization

Nearby neurons respond to similar features

Neuron can respond with intermediate rates to features deviating from maximum preference

Can be supported by local excitation

Lateral connections: surround suppression

Neuron can have suppressed response for features deviant from maximum preference

Typical V1 receptive field

Suppression/competition with interneurons

- In common cortical circuits, there are feedforward excitatory inputs and lateral inhibitory inputs

- Relative weighting achieves balance between activation and suppression

The pathway for smell processing

Nose/olfactory epithelium

Olfactory bulb (in cortex)

Orbitofrontal cortex

Amygdala

Hippocampus
Simplified circuit

- Olfactory Epithelium (OE) – input
- Mitral – output
- PGe – lateral inhibitor

Data in the brain

- Neural location related to information encoded
- Progression of encoding for increasingly complex concepts

Classes of representation

Local representation
- Neural level: “grandmother” cell
- “Region” level: face region, place region

Competition on behavior level

Opposing interpretations of scene

Data in the brain

- Neural location related to information encoded
- Progression of encoding for increasingly complex concepts

Classes of representation

Local representation
- Neural level: “grandmother” cell
- “Region” level: face region, place region

Competition on behavior level

Opposing interpretations of scene
Classes of representation

Fully distributed representation
- Every neuron/region plays a part

Sparsely-distributed representation
- Neural level: hyper-column for perceptual feature
- “Region” level: face network in medial temporal, lateral temporal, anterior parietal

Principles of information coding: binary

How many things can we represent with \( n \) binary (\( g^{\text{step}} \) activation function) neurons?
- Complete sparse coding: \( n \) things
- Complete distributed coding: \( 2^n \) things

Biology of sparse coding

- Preserving energy – higher spiking rate requires higher energy
- Representational fan-out
  - \(~1\) million neurons in retina \(~\rightarrow\) \(~140\) million neurons in V1 (primary visual cortex)
  - \(~50,000\) neurons in cochlea \(~\rightarrow\) \(~1.6\) million neurons in A1 (primary auditory cortex)

Coding on a scale: sparsity

Typically we will say “sparsity” is using at most 10% of available neurons
Coping on a scale: distributed + overlapping

· high firing
· mid firing
· not firing

mood: sad – happy
age: 0 – 100
amount hair: bald – long

What does this encode?

Coding on a scale: distributed + overlapping

Responses for each property add together

<table>
<thead>
<tr>
<th>mood</th>
<th>age</th>
<th>amount hair</th>
</tr>
</thead>
<tbody>
<tr>
<td>sad</td>
<td>0.1</td>
<td>0.1 – bald</td>
</tr>
<tr>
<td>neutral</td>
<td>0.5</td>
<td>0.5 – middle</td>
</tr>
<tr>
<td>happy</td>
<td>0.9</td>
<td>0.9 – full-hair</td>
</tr>
<tr>
<td>(sad – happy)</td>
<td>(0 – 100)</td>
<td>(bald – long)</td>
</tr>
</tbody>
</table>

How do we encode: sad (0), mid-age (.5), hairy (1.0)?

\[ \sum_j \text{level}_j \text{pattern}_j \]

n1 n2 n3
0 0 0
0 0.5
0 0 1
0.5 1.5

Coding on a scale: distributed + overlapping

Responses for each property add together

<table>
<thead>
<tr>
<th>mood</th>
<th>age</th>
<th>amount hair</th>
</tr>
</thead>
<tbody>
<tr>
<td>sad</td>
<td>0.1</td>
<td>0.1 – bald</td>
</tr>
<tr>
<td>neutral</td>
<td>0.5</td>
<td>0.5 – middle</td>
</tr>
<tr>
<td>happy</td>
<td>0.9</td>
<td>0.9 – full-hair</td>
</tr>
<tr>
<td>(sad – happy)</td>
<td>(0 – 100)</td>
<td>(bald – long)</td>
</tr>
</tbody>
</table>

How do we encode: happy-ish (.8), young-ish (.2), some-hair (0.5)?

\[ \sum_j \text{level}_j \text{pattern}_j \]

n1 n2 n3
0.8 0.8
0 0.2
0 0 0.5
0.9 0.2 1.5

Coding on a scale: distributed + overlapping

Responses for each property add together

<table>
<thead>
<tr>
<th>mood</th>
<th>age</th>
<th>amount hair</th>
</tr>
</thead>
<tbody>
<tr>
<td>sad</td>
<td>0.1</td>
<td>0.1 – bald</td>
</tr>
<tr>
<td>neutral</td>
<td>0.5</td>
<td>0.5 – middle</td>
</tr>
<tr>
<td>happy</td>
<td>0.9</td>
<td>0.9 – full-hair</td>
</tr>
<tr>
<td>(sad – happy)</td>
<td>(0 – 100)</td>
<td>(bald – long)</td>
</tr>
</tbody>
</table>

How do we encode: sad (0), mid-age (.5), hairy (1.0)?

\[ \sum_j \text{level}_j \text{pattern}_j \]

n1 n2 n3
0 0 0
0 0.5
0 0 1
0.5 1.5

What does this encode? 0.4 0.8

What does this encode? 1.5 1.5
Coding on a scale: distributed + overlapping

Responses for each property add together

<table>
<thead>
<tr>
<th>Mood</th>
<th>Age</th>
<th>Amount Hair</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.1</td>
<td>0.0</td>
</tr>
<tr>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>0.9</td>
<td>0.9</td>
<td>0.9</td>
</tr>
</tbody>
</table>

What does this encode? 0.4 0.8

Very sad: contributes: 0 \times [1 0 1] = 0 0 0
Middle age: contributes: 0.4 \times [0 1 1] = 0.4 0.4
Middle hair: contributes: 0.4 \times [0 0 1] = 0 0 0.4
Summing together: 0.4 0.8

Decoding with tuning curves

Use spiking rates from multiple neurons to determine encoded feature

- 15 Hz firing rate for red neuron means sound 400 or 800 Hz (at 10 dB)
- 15 Hz for red and 6 Hz for blue requires sound 800 Hz (at 10 dB)

Actual decoding incorporates noise/natural variability in spiking

Population coding to find direction of motion

Non-normalized population coding

$$S_{dir} = \sum_i r_i s_{pref}^i$$

$$s_{pref} = \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ -1 \end{bmatrix}$$

$$r = \begin{bmatrix} 1 \\ 4 \\ 1 \\ 0 \end{bmatrix}$$

$$S_{dir} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$
Population coding to find direction of motion

Non-normalized population coding

\[ s_{dir} = \sum_i r_i s_{pref} \]

```
 r | 1 4 1 0
----|----|----|----|
 S_{pref} | [ 0 ] | [ 1 ] | [ 0 ] | [ -1 ]
           | [ -1 ] | [ 0 ] | [ 1 ] | [ -1 ]
```

\[ \hat{r} = r - r_{\text{min}} \]

```
 r | 4 6 1
----|----|----|
 S_{pref} | [ 1 ] | [ 0 ]
```

Normalized firing rates

\[ r_{\text{min}} = 0 \, \text{Hz}, \, r_{\text{max}} = 60 \, \text{Hz} \]

```
 r | 30 30 10 0
----|----|----|----|
 S_{pref} | [ 0 ] | [ 1 ] | [ 0 ] | [ -1 ]
           | [ -1 ] | [ 0 ] | [ 1 ] | [ -1 ]
```

Population coding to find direction of motion

“Normalized” firing rate

\[ \hat{r}_i = \frac{r_i - r_{\text{min}}}{r_{\text{max}} - r_{\text{min}}} \]

If \( r_{\text{min}} = 1, \, r_{\text{max}} = 6 \) for

Then \( \hat{r}_i = \frac{4 - 1}{6 - 1} = \frac{3}{5} = 0.6 \).

Normalized \( \hat{r} \) will always be between 0 and 1

```
 r | 4
----|----|
 S_{pref} | [ 1 ] | [ 0 ]
```

Normalized firing rates

\[ r_{\text{min}} = 0 \, \text{Hz}, \, r_{\text{max}} = 60 \, \text{Hz} \]

```
 r | 0.5 0.5 0.16 0
----|----|----|----|
 r | 30 30 10 0
----|----|----|----|
 S_{pref} | [ 0 ] | [ 1 ] | [ 0 ] | [ -1 ]
           | [ -1 ] | [ 0 ] | [ 1 ] | [ -1 ]
```

\[ [0.5, 0] \]
Population coding to find direction of motion

"Normalized" pop'n coding

\[ \hat{s}_{\text{pop}} = \frac{\sum_i \hat{r}_i s_{i}^{\text{pref}}}{\sum_j \hat{r}_j} \]

<table>
<thead>
<tr>
<th>( \hat{r} )</th>
<th>0.05</th>
<th>0.5</th>
<th>0.05</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \hat{s}_{\text{pop}} )</td>
<td><img src="image1.png" alt="Diagram of arrows pointing in different directions" /></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Assume for all neurons \( r_{\text{min}} = 10 \text{ Hz}, r_{\text{max}} = 100 \text{ Hz} \)

\[
s_{\text{pref}} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix} \]

Another example

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{0.05}{0.6} \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{0.6} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{0.05}{0.6} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.83}{0} \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

Find most-favored motion direction, do not amplify motion distance

Population coding to find direction of motion

"Normalized" pop'n coding

\[ \hat{s}_{\text{pop}} = \frac{\sum_i \hat{r}_i s_{i}^{\text{pref}}}{\sum_j \hat{r}_j} \]

<table>
<thead>
<tr>
<th>( \hat{r} )</th>
<th>0.05</th>
<th>0.5</th>
<th>0.05</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \hat{s}_{\text{pop}} )</td>
<td><img src="image2.png" alt="Diagram of arrows pointing in different directions" /></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Assume for all neurons \( r_{\text{min}} = 10 \text{ Hz}, r_{\text{max}} = 100 \text{ Hz} \)

\[
s_{\text{pref}} = \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} -1 \\ 0 \end{bmatrix} \]

\[
\sum_j \hat{r}_j = 0.05 + 0.5 + 0.05 + 0 = 0.6
\]

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{0.05}{0.6} \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{0.6} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{0.05}{0.6} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.83}{0} \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

Find most-favored motion direction, do not amplify motion distance

Another example

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{0.05}{0.6} \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{0.6} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{0.05}{0.6} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.83}{0} \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

Another example

Assume for all neurons \( r_{\text{min}} = 10 \text{ Hz}, r_{\text{max}} = 100 \text{ Hz} \)

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{0.05}{0.6} \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{0.6} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{0.05}{0.6} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.83}{0} \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

Another example

Assume for all neurons \( r_{\text{min}} = 10 \text{ Hz}, r_{\text{max}} = 100 \text{ Hz} \)

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{0.05}{0.6} \begin{bmatrix} 0 \\ -1 \end{bmatrix} + \frac{0.5}{0.6} \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \frac{0.05}{0.6} \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{0.83}{0} \begin{bmatrix} -1 \\ 0 \end{bmatrix}
\]

\[ \hat{r}_{\text{pop}}^2 = \frac{0.4}{0.4} \frac{.6}{0.6} + \frac{0.2}{1.2} = 1.2 \]

\[ \hat{r}_{\text{pop}}^2 = \frac{0.33}{.33} \]

\[ \hat{s}_{\text{pop}} = \begin{bmatrix} 0.34 \\ -0.33 \end{bmatrix} \]

\[ \hat{s}_{\text{pop}}^2 = 0.33 \]
A third example

Assume for all neurons $r_{\text{min}}=20 \text{ Hz}$, $r_{\text{max}}=80 \text{ Hz}$

<table>
<thead>
<tr>
<th>$r$</th>
<th>20</th>
<th>20</th>
<th>30</th>
<th>50</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/150" alt="image" /></td>
<td><img src="https://via.placeholder.com/150" alt="image" /></td>
<td><img src="https://via.placeholder.com/150" alt="image" /></td>
<td><img src="https://via.placeholder.com/150" alt="image" /></td>
<td></td>
</tr>
</tbody>
</table>

$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix}$

$\begin{array}{cccc}
\frac{20-20}{80} & \frac{20-20}{80} & \frac{30-20}{80} & \frac{50-20}{80} \\
\hat{p} & 0 & 0 & 0.13 & 0.38 \\
\hat{p}_{\text{pop}} & 0 & 0 & 0.13 & 0.38 \\
\hat{s}_{\text{pop}} & 0.51 & 0.51 & 0.13 & 0.38 \\
\end{array}$

Decoding large neural codes

Information from neuron patterns
- Happy
- Old
- Hairy
- Loud

Overlay of multiple patterns and noise
- What property is this?

Decoding large neural codes

Classifier:
- If consistent response, can learn pattern
- If irrelevant response, cannot learn helpful pattern

Method:
- 500 trials – measure mood, record brain responses
- Make classifier from neural patterns in trials 1-250
- Find accuracy to predict mood in trials 251-500