Matlab, part 4:
Vector analysis

Matrix math

\[
\begin{bmatrix}
\vdots \\
v_1 & v_2 & v_3 \\
\vdots \\
v_1 & v_2 & v_3 \\
\vdots \\
v_1 & v_2 & v_3 \\
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
z \\
\end{bmatrix} = xv_1 + yv_2 + zv_3
\]

Assuming right matrix is a single column
In general, # of left matrix columns must equal # of right matrix rows

Matrices and weighted sums

\[
\begin{bmatrix}
1 & 4 & 1 & 0 \\
0 & 1 & 0 & 0 \\
-1 & 0 & 0 & 1 \\
0 & -1 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
1 \\
0 \\
1 \\
0 \\
\end{bmatrix}
= \begin{bmatrix}
4 \\
0 \\
1 \\
0 \\
\end{bmatrix}
\]

Left Matrix columns times Right matrix numbers

\[
\begin{bmatrix}
x \\
y \\
z \\
\end{bmatrix} = \begin{bmatrix}
xv_1 + yv_2 + zv_3 \\
\vdots \\
\vdots \\
\vdots \\
\end{bmatrix}
\]

Matrix math

\[
A = [1 2; 3 4];
b = [4; 5];
\]

What is \(A*b\)?
\[
[19; 28]
\]

Transpose:
\[
[4; 5] == [4 5]'
\]

\(a'\) flips rows and columns
Scatter-plots
Visualizing how two variables vary together

<table>
<thead>
<tr>
<th>Reaction time</th>
<th>Cortical response</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10</td>
</tr>
<tr>
<td>45</td>
<td>25</td>
</tr>
<tr>
<td>150</td>
<td>6</td>
</tr>
<tr>
<td>30</td>
<td>15</td>
</tr>
<tr>
<td>125</td>
<td>14</td>
</tr>
<tr>
<td>33</td>
<td>12</td>
</tr>
<tr>
<td>200</td>
<td>3</td>
</tr>
</tbody>
</table>

plot(var1, var2, '.')
scatter(var1, var2)

Comparing vectors
Can compare 2 vectors
- by multiplying \( a \times b' \)
  - high product = high similarity
- by correlating \( \text{corr}(a, b) \)
  - between -1 and 1
  - high |correlation| = high connection between vectors

LOC localizer: experimental design
Each second:
- new object OR
- new noise OR
- “blank screen” (fixation)

Visual comparison
timesObjs2, at each second:
- 0 for no-object
- 1 for yes-object
Voxel response
\( \text{neuroData2}(24, 26, 4,:) \)
at each second neural response to stimuli
Numeric comparison

Single voxel response:
voxRes1=squeeze(neuroData2(24,26,4,:));

Compare with object appearance times:
corr(voxRes1, timesObjs2');

Consider correlations at multiple locations (axial slice):
for x=1:32,
   for y=1:32
      voxRes=squeeze(neuroData2(x,y,5));
      corrMat(x,y)=corr(voxRes,timesObjs2');
   end
end