HW3Q2

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import chi2_contingency
from mpl_toolkits.mplot3d import Axes3D
import seaborn as sns
```

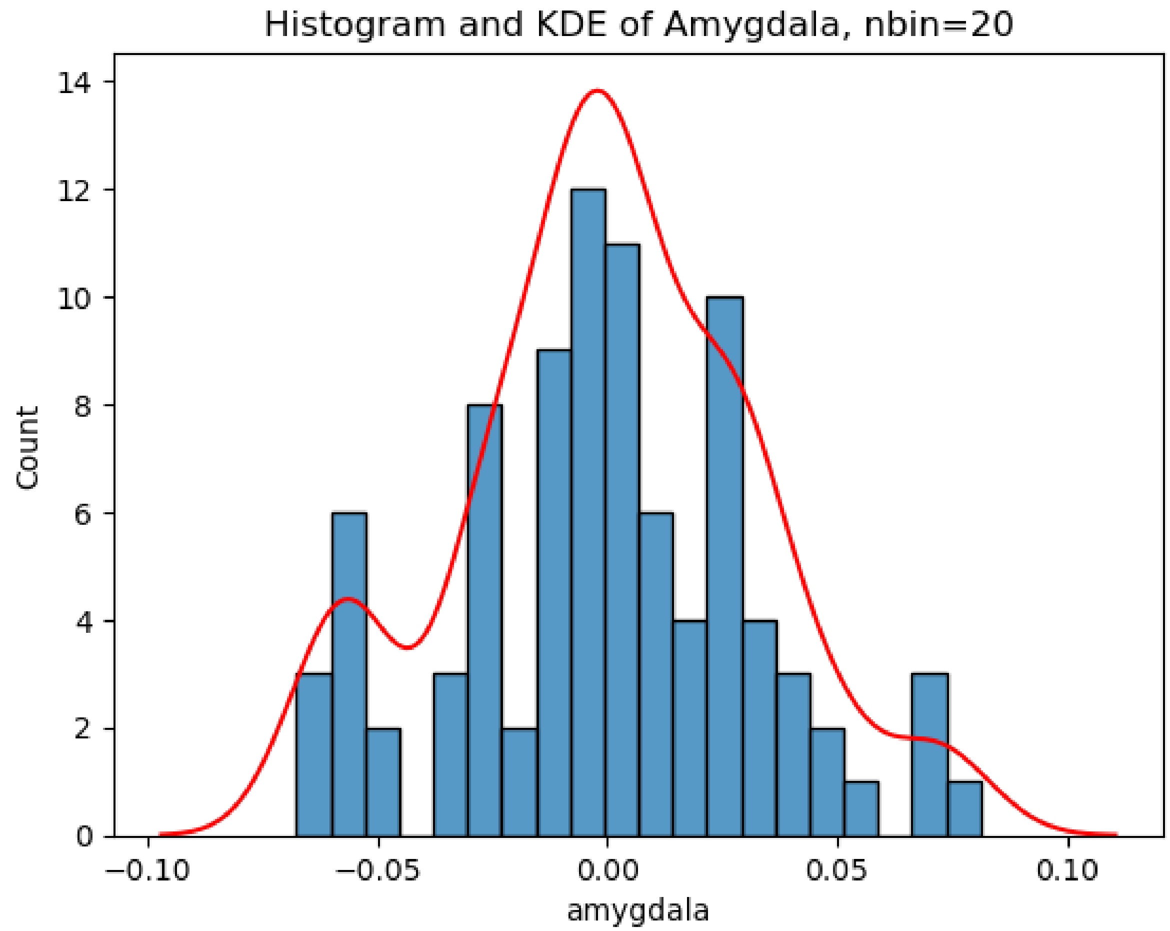
In [2]:

```
path = "data/n90pol.csv"
df = pd.read_csv(path)
data = df[['amygdala', 'acc']]
```

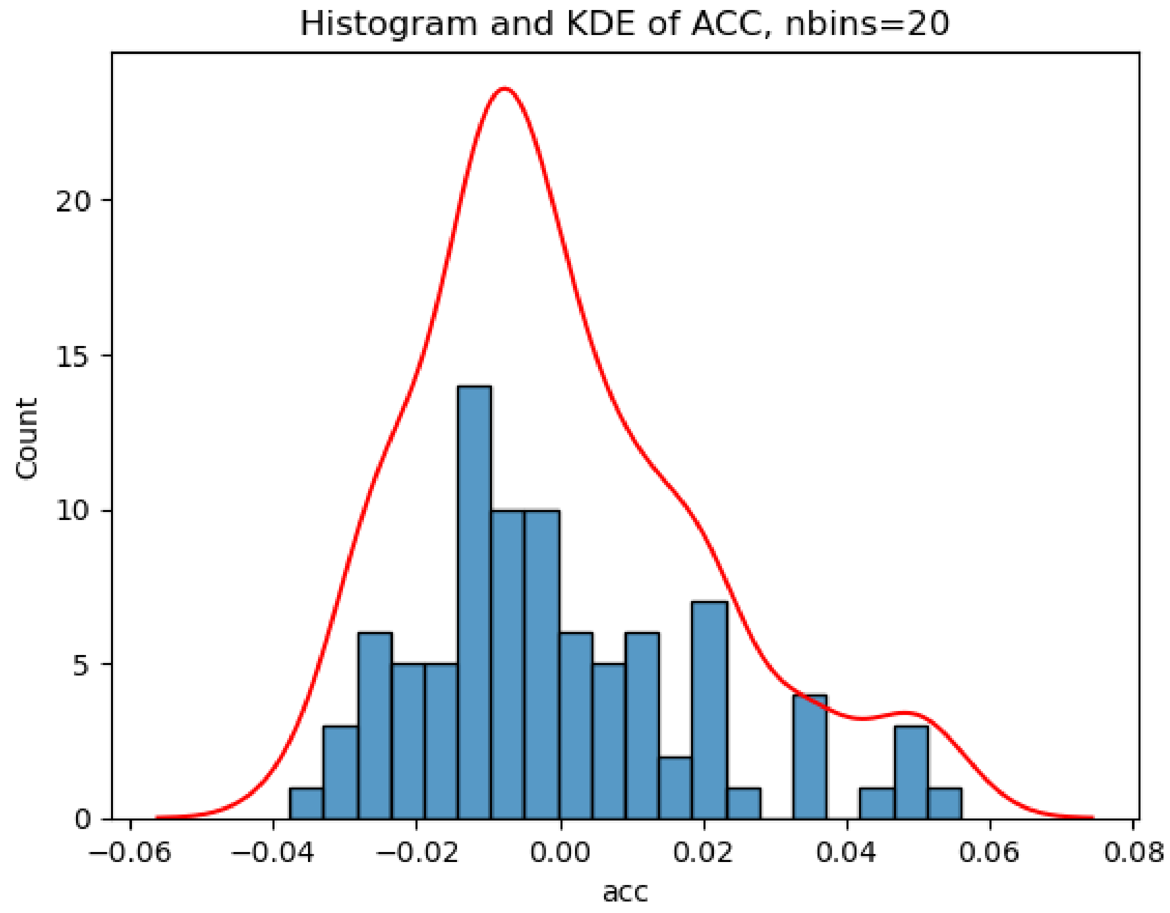
(a)

In [3]:

```
# Amygdala
sns.histplot(df['amygdala'], bins=20)
sns.kdeplot(df['amygdala'], color='r', bw_method=0.3)
plt.title("Histogram and KDE of Amygdala, nbin=20")
#plt.savefig('Histogram and KDE of Amygdala, nbin=20')
plt.show()

# ACC
sns.histplot(df['acc'], bins=20)
sns.kdeplot(df['acc'], color='r', bw_method=0.3)
plt.title("Histogram and KDE of ACC, nbins=20")
#plt.savefig('Histogram and KDE of ACC, nbins=20')
plt.show()
```



Histogram and KDE of Amygdala, nbin=20

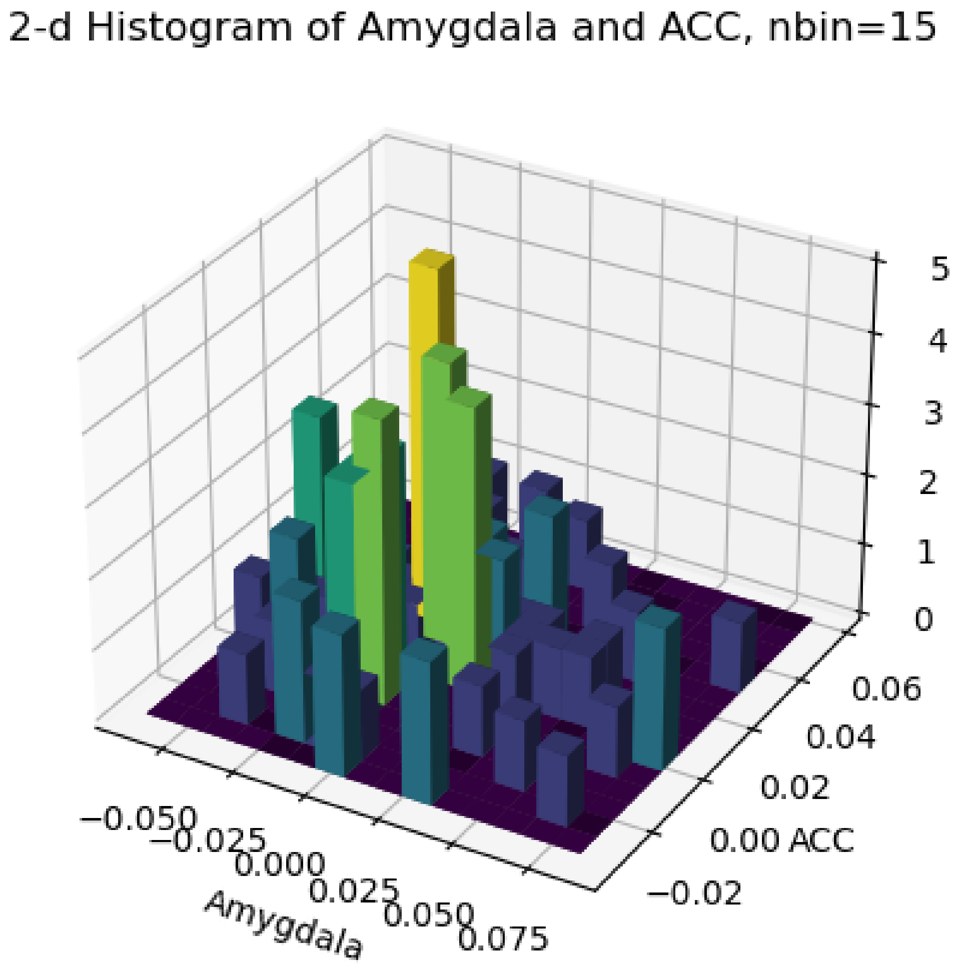


Histogram and KDE of ACC, nbins=20

(b)

In [4]:

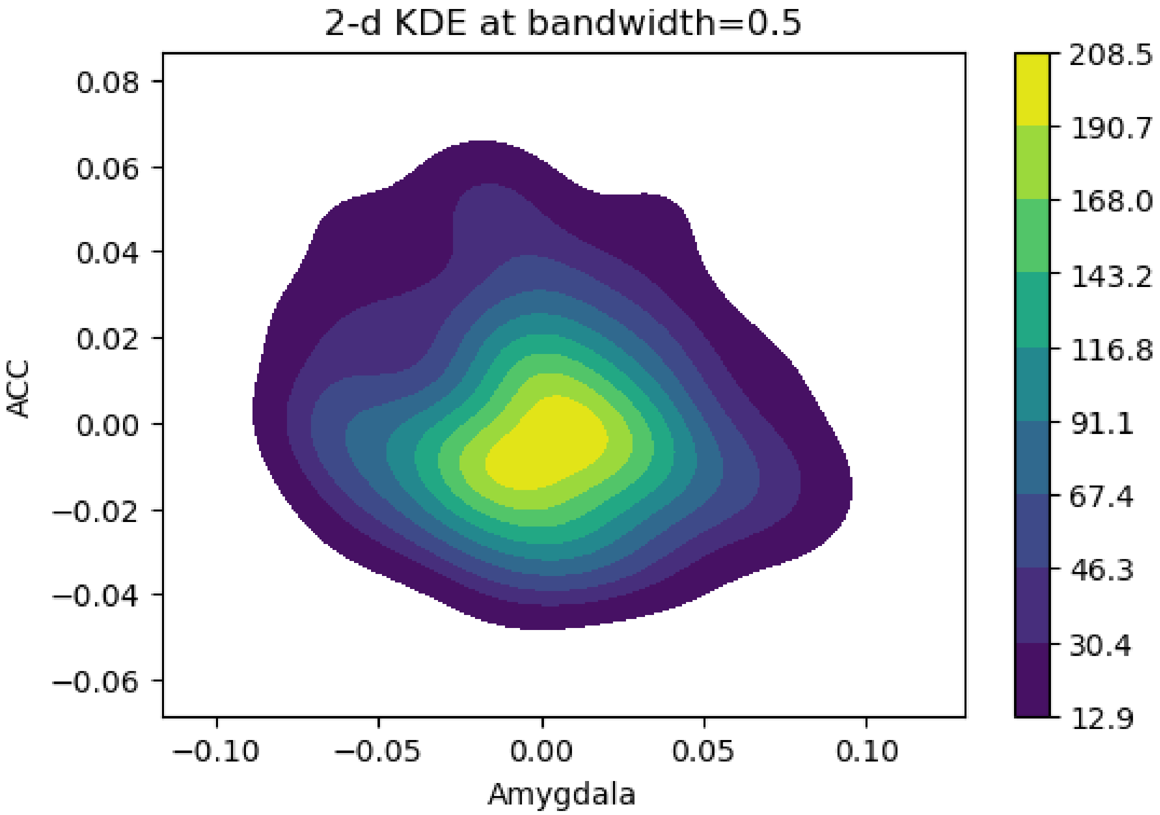
```
#2-d histogram
min_data = data.min().values
max_data = data.max().values
nbin = 15

fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
hist, xedges, yedges = np.histogram2d(df['amygdala'], df['acc']

xpos, ypos = np.meshgrid(xedges[:-1] + xedges[1:], yedges[:-1]
xpos = xpos.flatten()/2.
ypos = ypos.flatten()/2.

zpos = np.zeros_like(xpos)
dx = xedges[1] - xedges[0]
dy = yedges[1] - yedges[0]
dz = hist.flatten()

normalize = plt.Normalize(dz.min(), dz.max())
ax.bar3d(xpos, ypos, zpos, dx, dy, dz, color=plt.cm.viridis(nor
ax.set_xlabel('Amygdala')
ax.set_ylabel('ACC')
ax.set_title("2-d Histogram of Amygdala and ACC, nbin=15")

#plt.savefig('2-d Histogram of Amygdala and ACC, nbin=15')
plt.show()
```



(c)

In [5]:

```
#2d heat-map
plt.figure(figsize=(6, 4))

kde_x = df['amygdala']
kde_y = df['acc']

sns.kdeplot(x=kde_x, y=kde_y, cmap='viridis', fill=True, cbar=T

plt.xlabel('Amygdala')
plt.ylabel('ACC')
plt.title("2-d KDE at bandwidth=0.5")
#plt.savefig("2d kde plot")
plt.show()
```



In [6]:

```
#chi2 test for independence
c_table = pd.crosstab(kde_x, kde_y)

chi2 = chi2_contingency(c_table)
print(chi2.pvalue)
```

```
0.30049182632409444
```

In [7]:

```
#covariance matrix for independence
covariance = np.cov(kde_x, kde_y)[0, 1]
print(covariance)
```

```
-8.560931960049936e-05
```

(d)

In [8]:

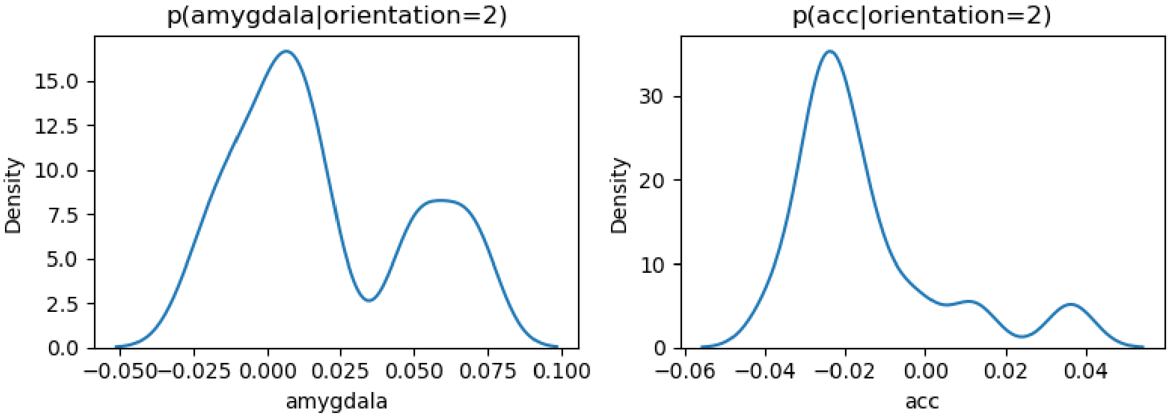
```
#calc conditional sample means
means = df.groupby('orientation')[['amygdala', 'acc']].mean()
print(means.T)
```

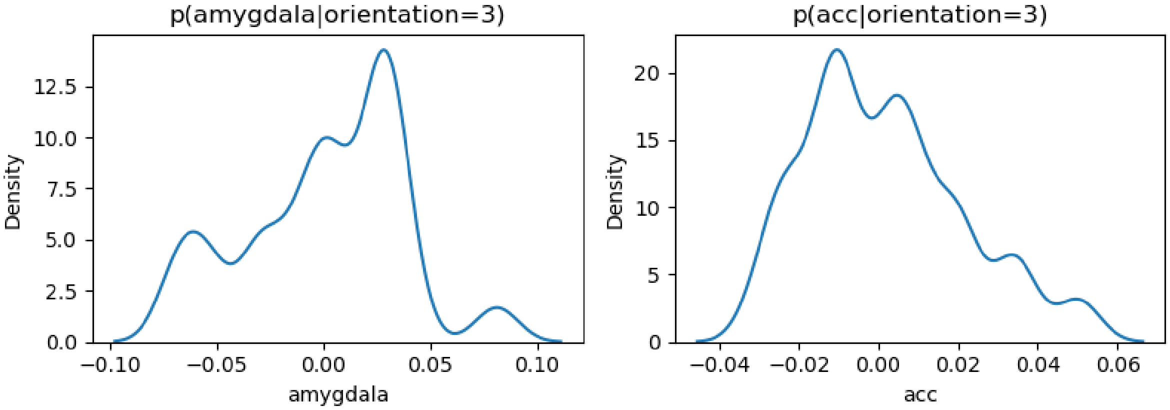
```
orientation         2         3        4         5
amygdala     0.019062  0.000588 -0.00472 -0.005692
acc         -0.014769  0.001671  0.00131  0.008142
```

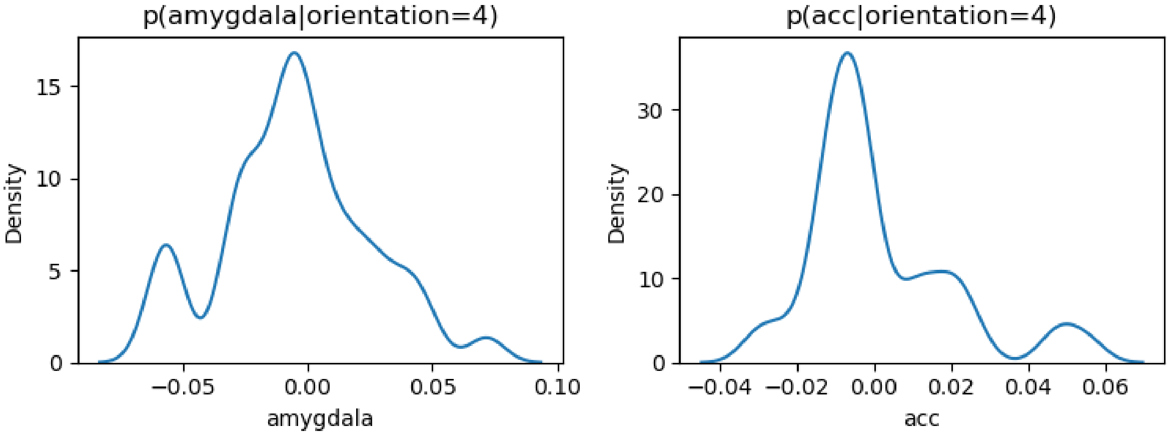
In [9]:

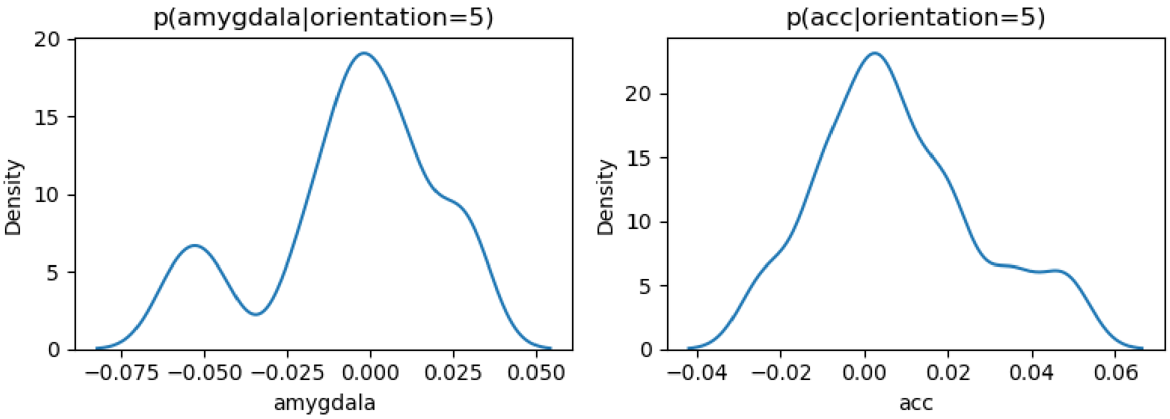
```
#plotting conditional distribution for c=2 to 6
for c in range(2, 6):
    orientation = df[df['orientation'] == c]
```

```
fig, axs = plt.subplots(1, 2, figsize=(8, 3))

sns.kdeplot(orientation['amygdala'], bw_adjust=0.5, ax=axs
axs[0].set_title(f"p(amygdala|orientation={c})")

sns.kdeplot(orientation['acc'], bw_adjust=0.5, ax=axs[1])
axs[1].set_title(f"p(acc|orientation={c})")

plt.tight_layout()
#plt.savefig(f"conditional distribution {c}")
plt.show()
```





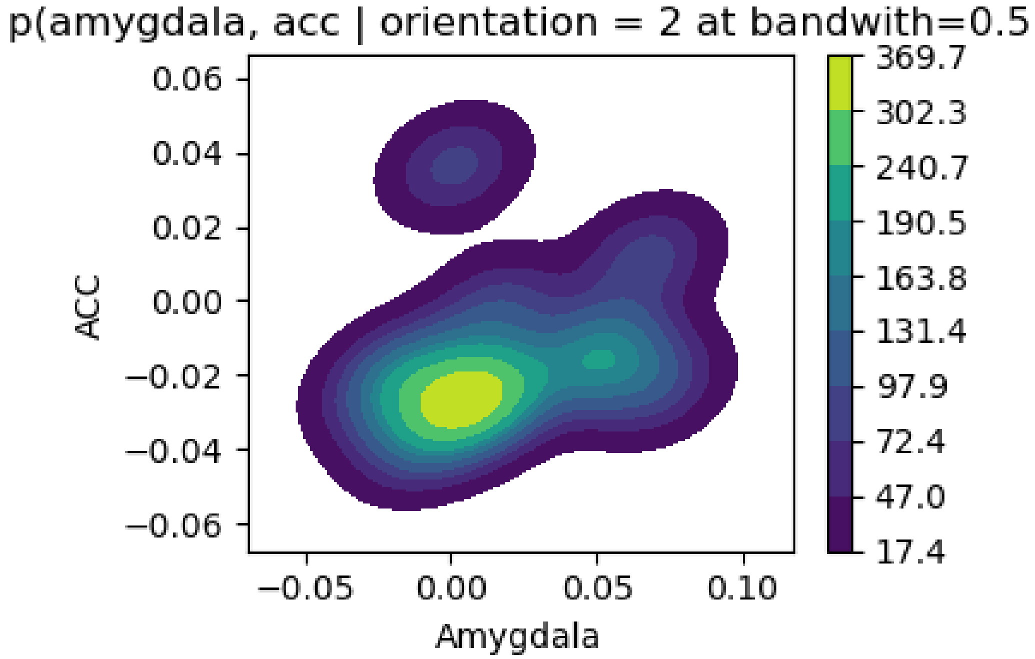


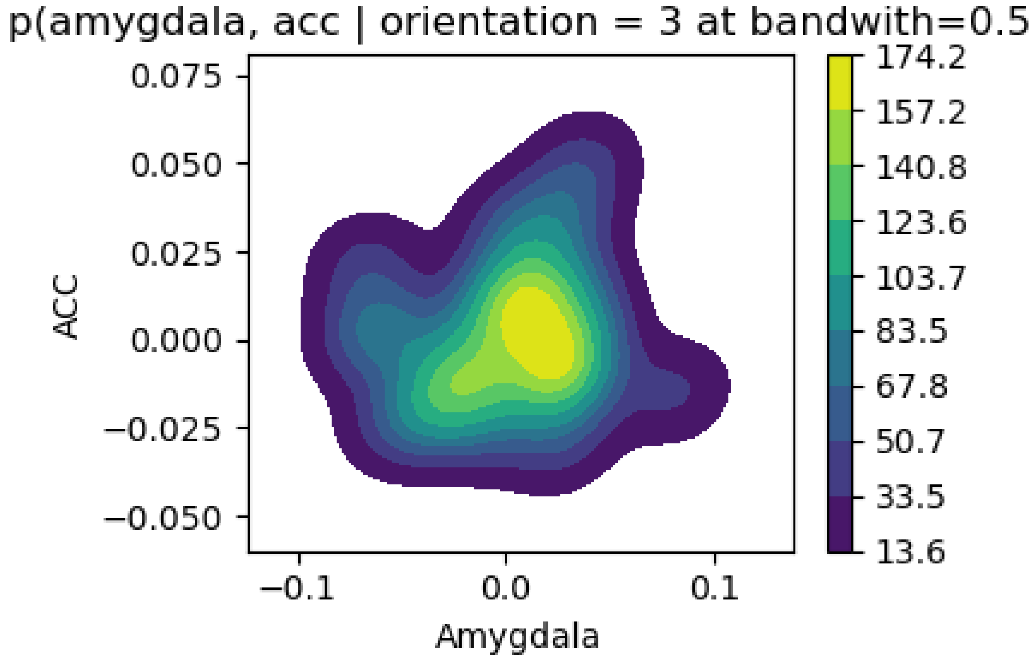


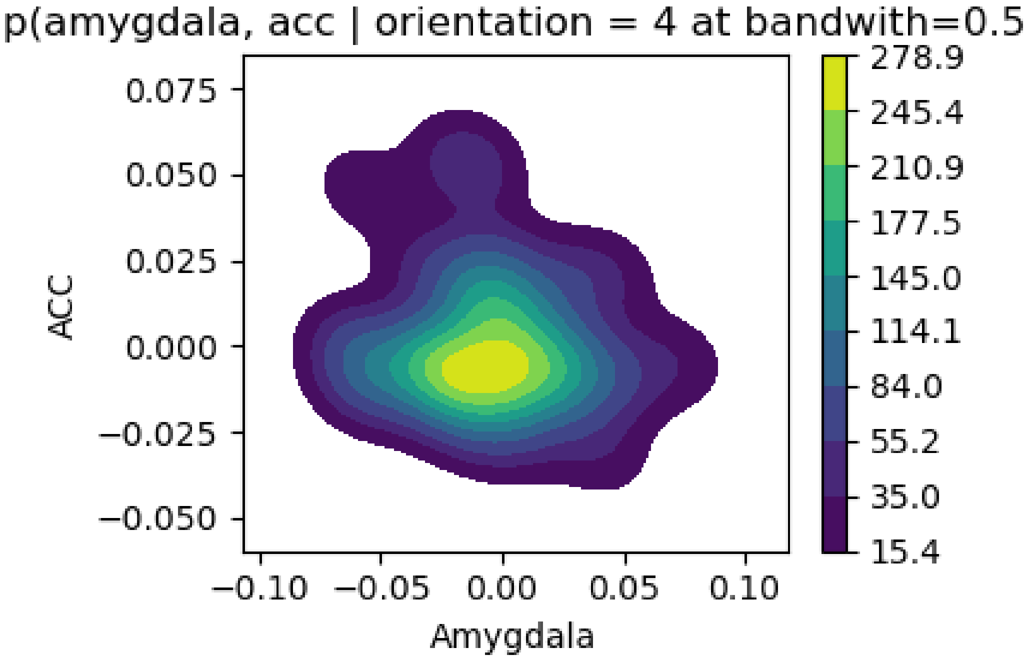
(e)

In [10]:

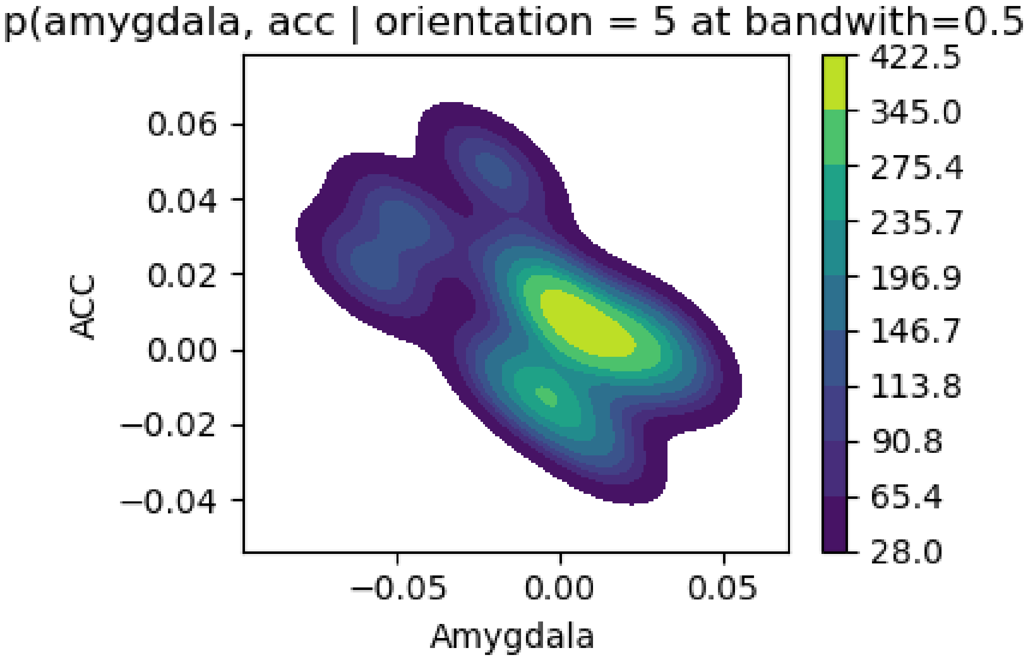
```
# Plotting joint distribution for each orientation
for c in range(2, 6):
    orientation = df[df['orientation'] == c]

    fig, ax = plt.subplots(1, 1, figsize=(4, 3))

    kde_x = orientation['amygdala']
    kde_y = orientation['acc']
    sns.kdeplot(x=kde_x, y=kde_y, bw_method=0.5, cmap='viridis'

    ax.set_title(f"p(amygdala, acc | orientation = {c} at bandw
    ax.set_xlabel('Amygdala')
    ax.set_ylabel('ACC')

    plt.tight_layout()
    #plt.savefig(f"joint_distribution_orientation_{c}.png")
    plt.show()
```


p(amygdala, acc | orientation = 2 at bandwith=0.5
ACC
0.06
0.04
0.02
0.00
−0.02
−0.04
−0.06
−0.05
0.00
0.05
0.10
Amygdala
369.7
302.3
240.7
190.5
163.8
131.4
97.9
72.4
47.0
17.4


p(amygdala, acc | orientation = 3 at bandwith=0.5
ACC
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.1
0.0
0.1
Amygdala
174.2
157.2
140.8
123.6
103.7
83.5
67.8
50.7
33.5
13.6



p(amygdala, acc | orientation = 4 at bandwith=0.5



p(amygdala, acc | orientation = 5 at bandwith=0.5