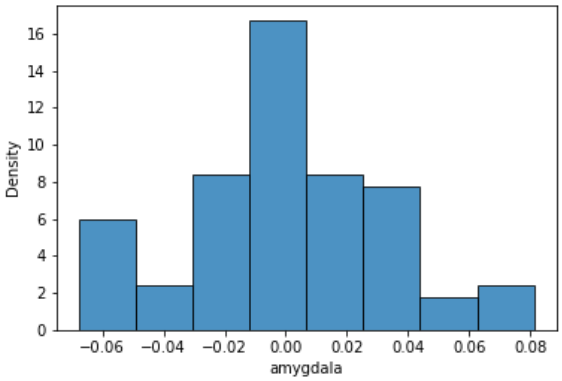
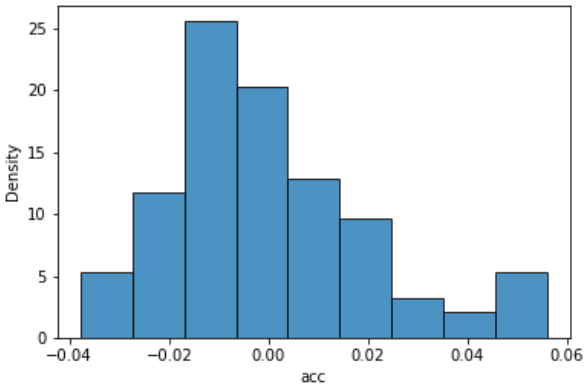
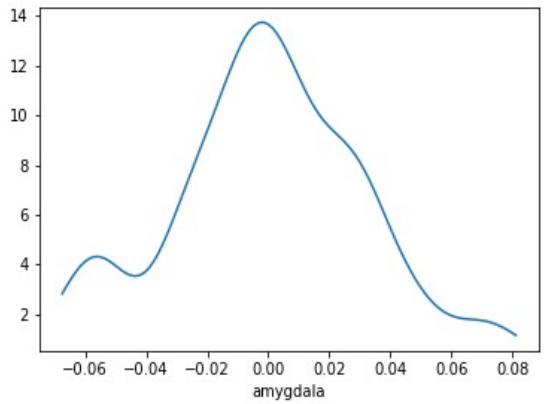
Richard Albright
ISYE-6740
HW3
Fall 2020

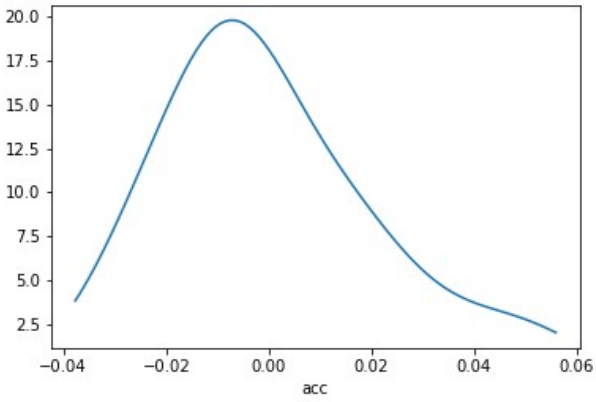
1. Density estimation: Psychological experiments.

a. Form the 1-dimensional histogram and KDE to estimate the distributions of amygdala and acc, respectively. For this question, you can ignore the variable orientation.

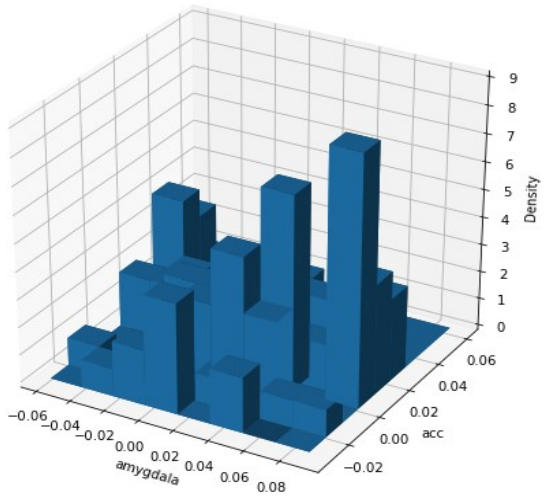


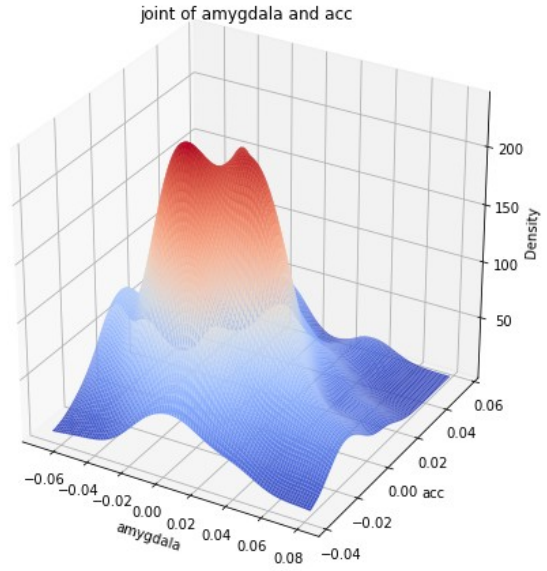




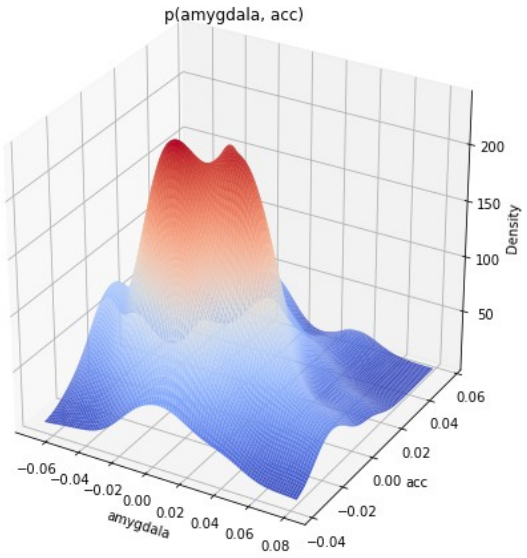


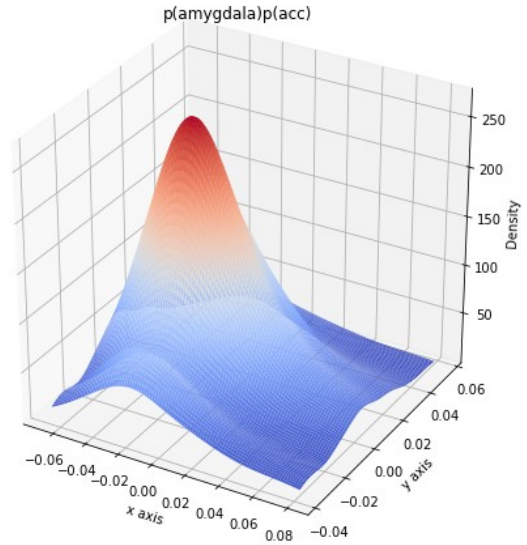
b. Form 2-dimensional histogram for the pairs of variables (amygdala,acc).Decide on a suitable number of bins so you can see the shape of the distribution clearly.Also use kernel-density-estimation (KDE) to estimate the 2-dimensional density func-tion of (amygdala,acc).

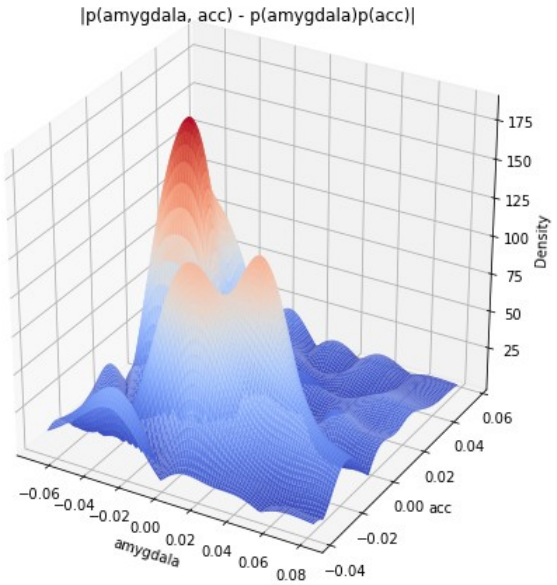




c. Using (a) and (b), using KDE estimators, verify whether or not the variables amygdala and acc are independent? You can tell this by checking do we approximately have p(amygdala,acc) =p(amygdala)p(acc)? To verify this, please show three plots: the map for p(amygdala,acc), the map for p(amygdala)p(acc) and the error map |p(amygdala,acc)−p(amygda la)p(acc)|. Comment on your results and whether this helps us to find out whether the two parts of brains (for emotions and decision-making)functions independently or they are related.

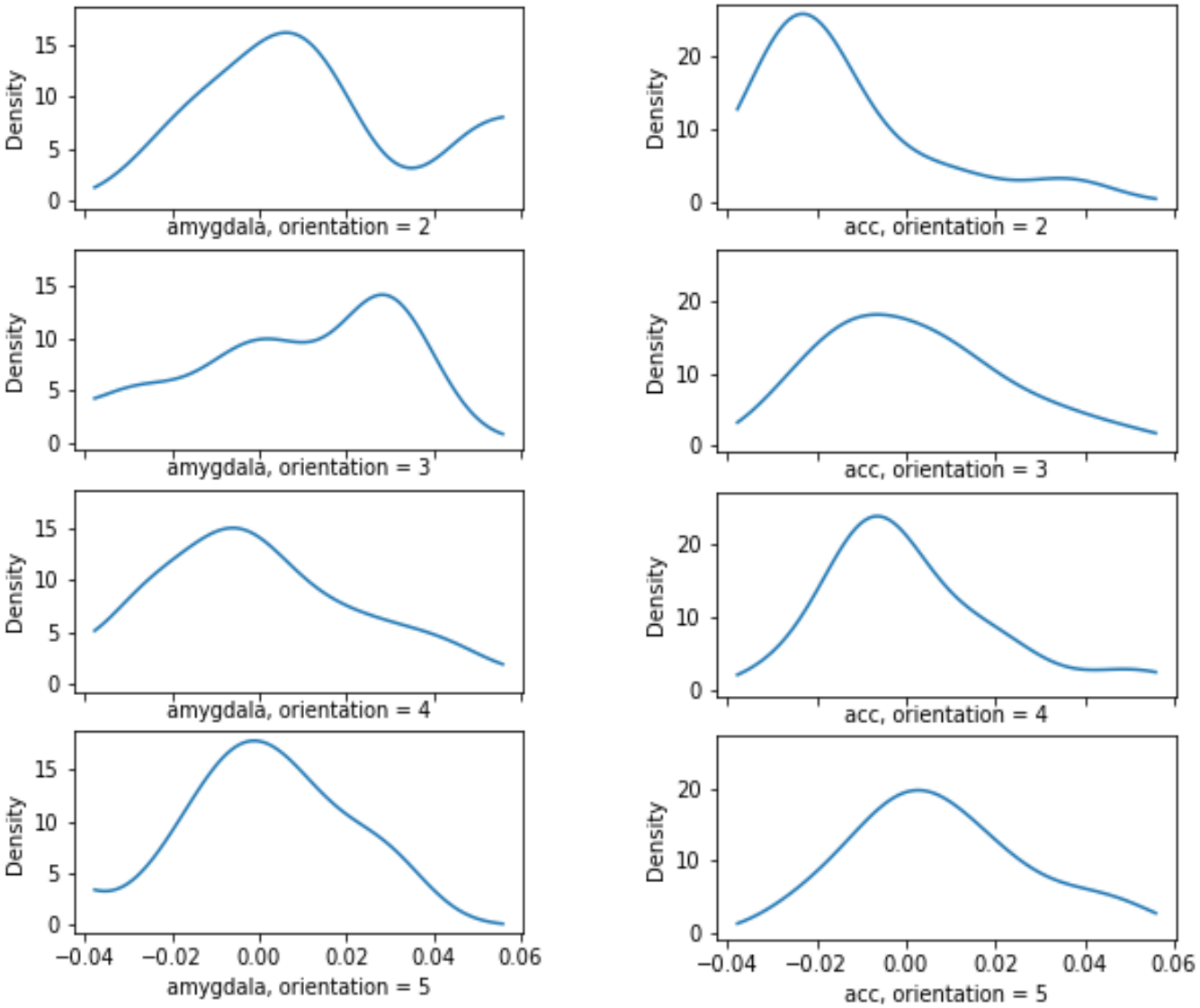




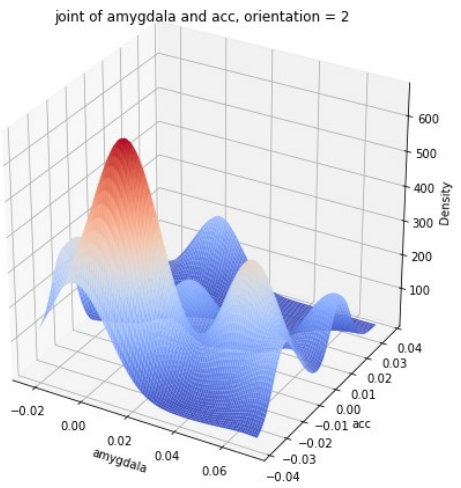


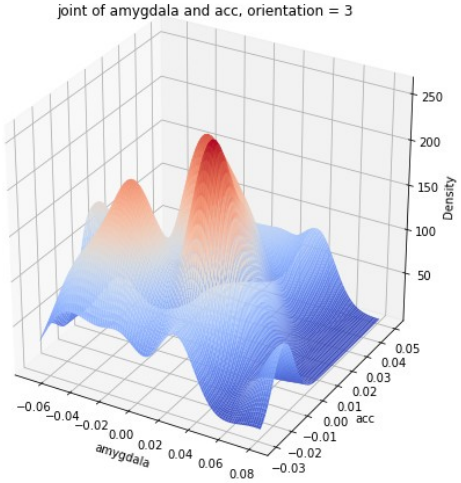
If amygdala and acc are independent, the joint distribution should roughly equal the product of the 2 distributions, and absolute error term would therefore be relatively flat. In this case, amygdala and acc are not independent since the absolute value of the error term is not flat, There are 2 peaks in the chart of the error term along both the axis of amygdala and acc.

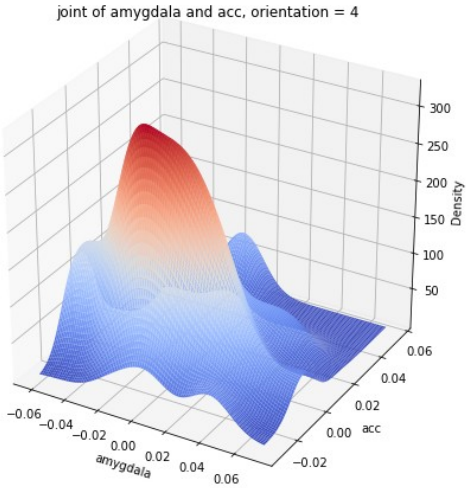
d. Now we will consider the variable orientation. We will estimate the conditional distribution of the volume of the amygdala, conditioning on political orientation: p(amygdala|orientation=c),c= 2,...,5. Do the same for the volume of the acc: Plot p(acc|orientation=c),c= 2,...,5. You will use KDE to achieve the goal. (Note that the conditional distribution can be understood as fitting a distribution for the data with the same (fixed) orientation. Thus there should be 4 one-dimensional distribution functions to show for this question.)

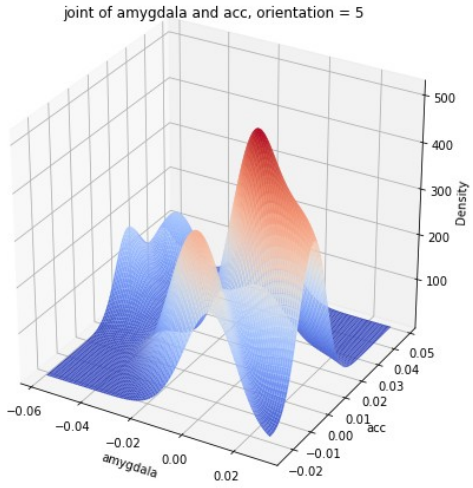


e. Again we will consider the variable orientation. We will estimate the conditional joint distribution of the volume of the amygdala and acc, conditioning on a function of political orientation:p(amygdala, acc|orientation=c),c= 2,...,5. You will use two-dimensional KDE to achieve the goal.

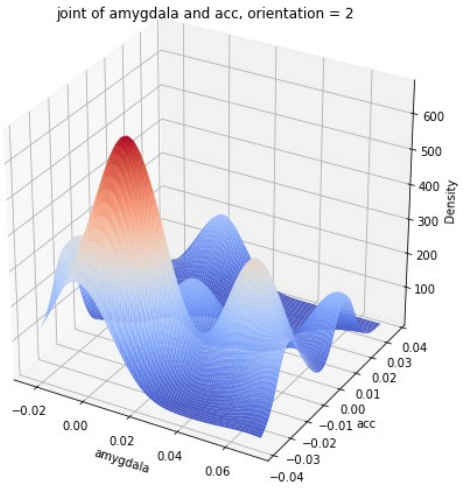


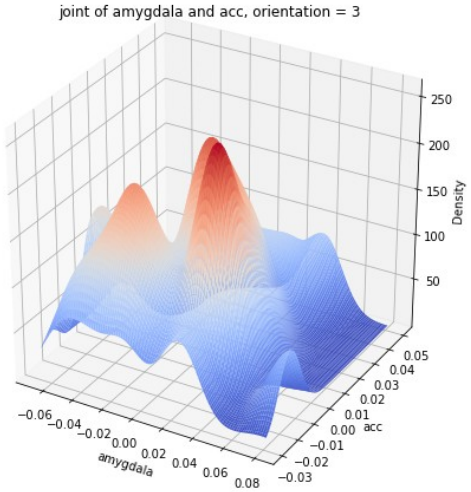


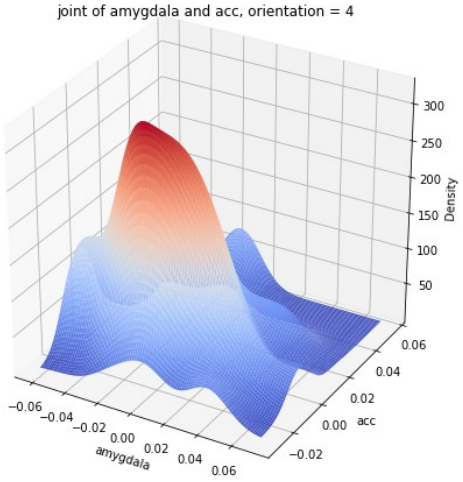


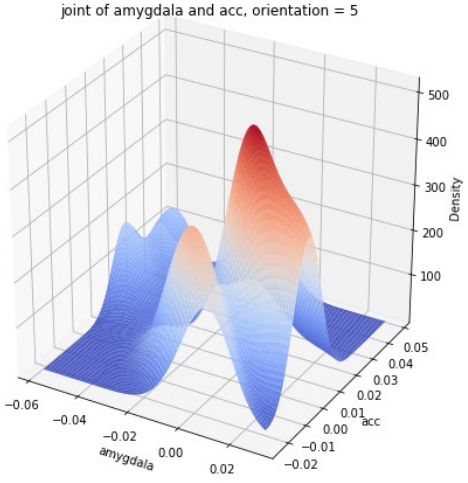


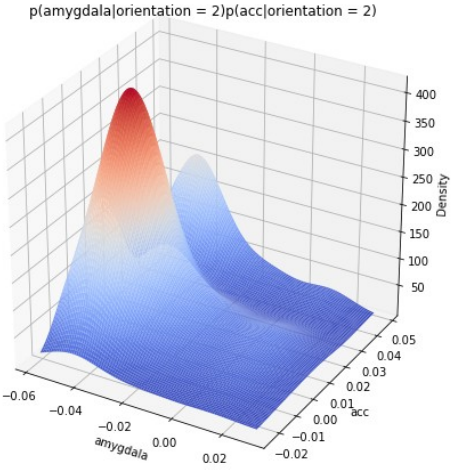
f. Using (d) and (e), evaluate whether or not the two variables are likely to be conditionally independent. To verify this, please show three plots: the map for p(amygdala,acc|orientation=c),the map for p(amygdala|orientation=c)p(acc|orientation=c) and the error map |p(amygdala,acc|orientation=c)−p(amygdala|orientation=c)p(acc|orientation=c)|, c= 2,...,5.Comment on your results and whether this helps us to find out whether the two parts of brains (for emotions and decision-making) functions independently or they are related, conditionally on the political orientation (i.e., considering different types of personality)

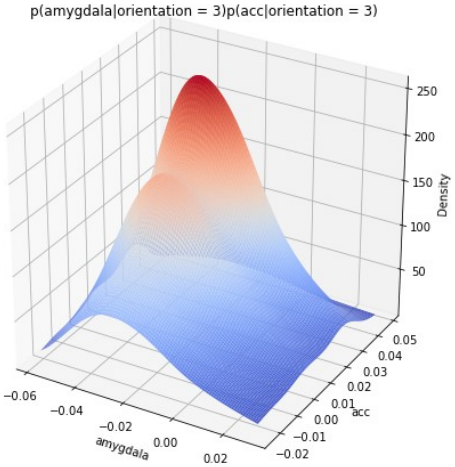


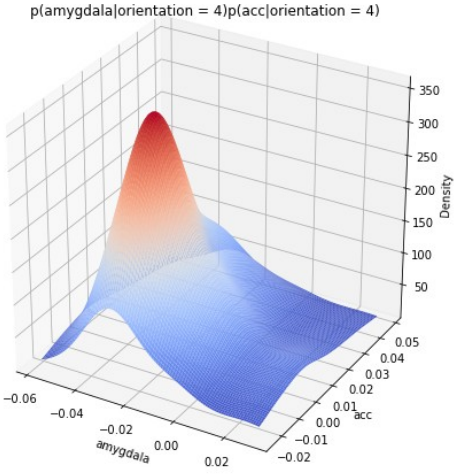


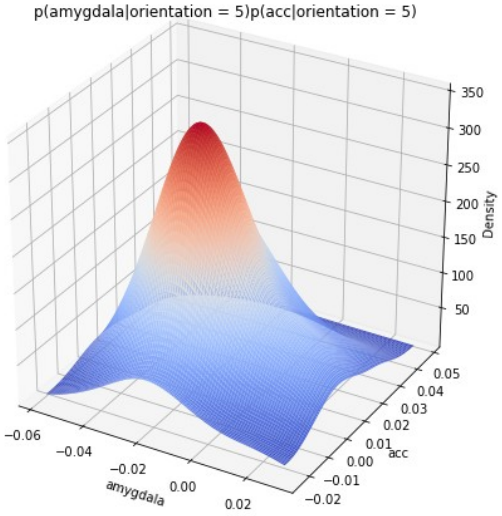


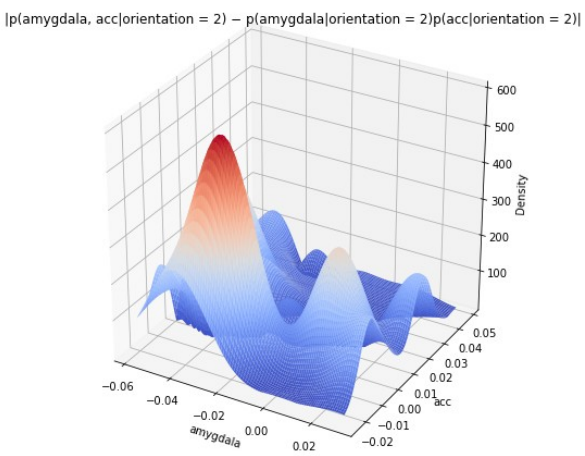


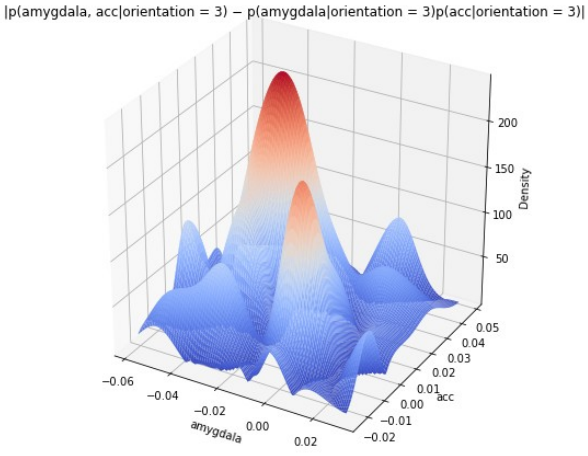

p(amygdala|orientation = 2)p(acc|orientation = 2)
Density
amygdala
acc

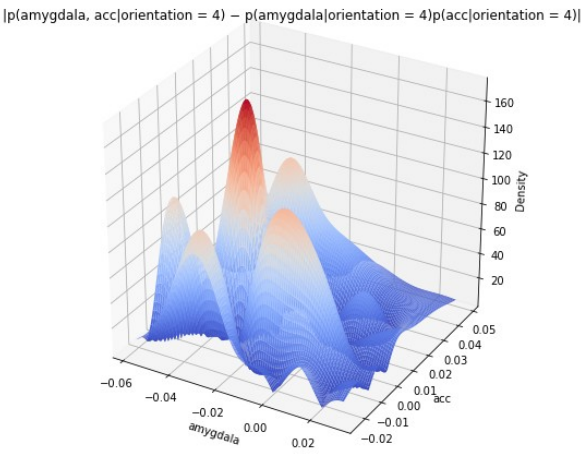

p(amygdala|orientation = 3)p(acc|orientation = 3)
Density
amygdala
acc

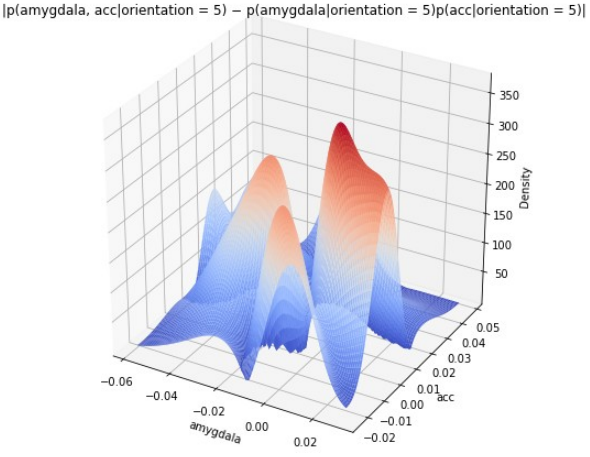

p(amygdala|orientation = 4)p(acc|orientation = 4)
Density
amygdala
acc


p(amygdala|orientation = 5)p(acc|orientation = 5)
Density
amygdala
acc









Conditionally, there is still a relationship between amygdala and acc for all personality types. The peaks for each personality type occur in different regions of the axes' between amygdala and acc for each different political orientation. The error is also significant those same regions.

2. Implementing EM for MNIST dataset, with PCA for dimensionality reduction.

a. Select from data one raw image of “2” and “6” and visualize them, respectively.

b. Write down detailed expression of the E-step and M-step in the EM algorithm (hint: when computing τ_{ik}, you can drop the $(2\pi)^{n/2}$ factor from the numerator and denominator expression, since it will be canceled out; this can help avoid some numerical issues in computation).

E-step

$$p(\mathbf{x};\phi,\mu,\Sigma) = \sum_{j=1}^{M} p(\mathbf{z}^{(j)}) p(\mathbf{x}|\mathbf{z}^{(j)};\mu,\Sigma)$$

$$\sum_{j=1}^{M} p(\mathbf{z}^{(j)}) = 1$$

$$Q^{(i)}(\mathbf{z}^{(j)}) = f(\theta) = \frac{p(\mathbf{x}^{(i)}|\mathbf{z}^{(j)};\mu,\Sigma)p(\mathbf{z}^{(j)})}{\sum_{k=1}^{M} p(\mathbf{x}^{(i)}|\mathbf{z}^{(k)};\mu,\Sigma)p(\mathbf{z}^{(k)})}$$

$$= \frac{\phi_j \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma_j|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_j)^T\Sigma_j^{-1}(\mathbf{x}^{(i)}-\mu_j)\}}{\sum_{k=1}^{M} \phi_k \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma_k|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\mathbf{x}^{(i)}-\mu_k)^T\Sigma_k^{-1}(\mathbf{x}^{(i)}-\mu_k)\}}$$

M-step

$q_{i,j} = Q^{(i)}(\mathbf{z}^{(j)})$

$$\frac{\partial L}{\partial \mu_k} = \frac{\partial}{\partial \mu_k}\Big[\sum_{i=1}^{n} -\frac{1}{2} q_{i,k}(\mathbf{x}^{(i)} - \mu_k)^T \Sigma_k^{-1}(\mathbf{x}^{(i)} - \mu_k)\Big]$$

$$= -\sum_{i=1}^{n} q_{i,k}\Sigma_k^{-1}(\mu_k - \mathbf{x}^{(i)})$$

$$= -\Sigma_k^{-1}(\sum_{i=1}^{n} q_{i,k}\mu_k - \sum_{i=1}^{n} q_{i,k}\mathbf{x}^{(i)}) = 0$$

$$\mu_k^{new} = \frac{\sum_{i=1}^{n} q_{i,k}\mathbf{x}^{(i)}}{\sum_{i=1}^{n} q_{i,k}},\ k = 1, \ldots, M$$

let

$\Lambda_k = \Sigma_k^{-1}$

$$\frac{\partial L}{\partial \Lambda_k} = \frac{\partial}{\partial \Lambda_k}\Big\{\sum_{i=1}^{n} q_{i,k}\Big[\frac{1}{2}\log|\Lambda_k| - \frac{1}{2}(\mathbf{x}^{(i)} - \mu_k)^T\Lambda_k(\mathbf{x}^{(i)} - \mu_k)\Big]\Big\}$$

$$= \frac{\partial}{\partial \Lambda_k}\Big\{\sum_{i=1}^{n} q_{i,k}\Big[\frac{1}{2}\log|\Lambda_k| - \frac{1}{2}\mathrm{tr}\Big((\mathbf{x}^{(i)} - \mu_k)^T\Lambda_k(\mathbf{x}^{(i)} - \mu_k)\Big)\Big]\Big\}$$

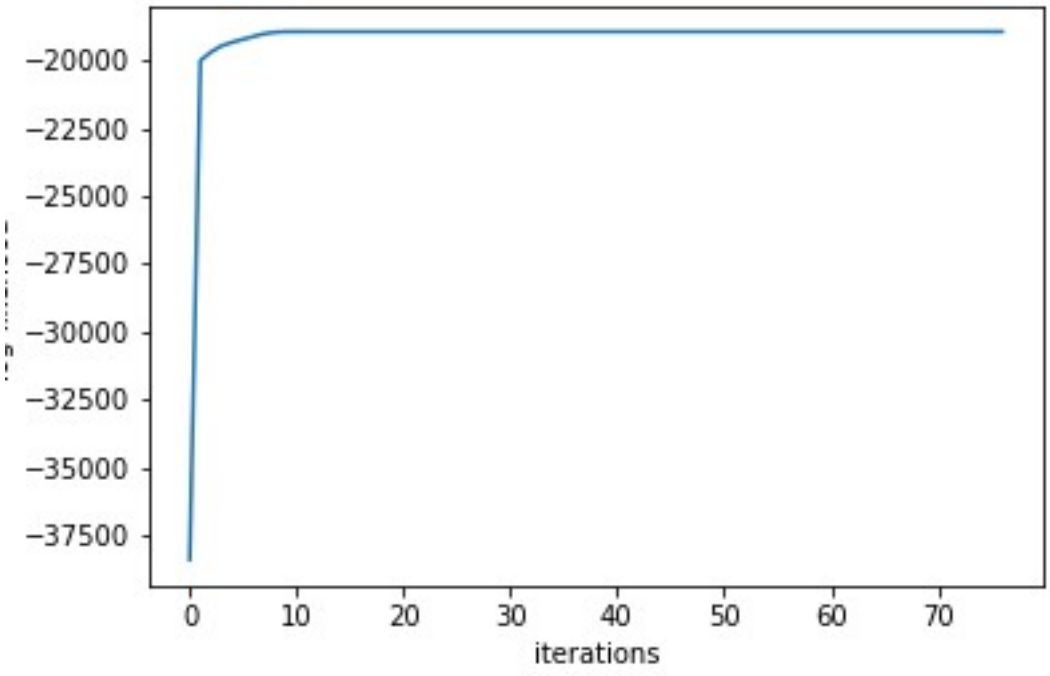
$$= \frac{1}{2}\sum_{i=1}^{n} q_{i,k}\Lambda_k^{-1} - \frac{1}{2}\frac{\partial}{\partial \Lambda_k}\mathrm{tr}\left(\Lambda_k\sum_{i=1}^{n} q_{i,k}(\mathbf{x}^{(i)} - \mu_k)(\mathbf{x}^{(i)} - \mu_k)^T\right)$$

$$= \frac{1}{2}\sum_{i=1}^{n} q_{i,k}\Sigma_k - \frac{1}{2}\sum_{i=1}^{n} q_{i,k}(\mathbf{x}^{(i)} - \mu_k)(\mathbf{x}^{(i)} - \mu_k)^T = 0$$

$$\Sigma_k^{\mathrm{new}} = \frac{\sum_{i=1}^{n} q_{i,k}(\mathbf{x}^{(i)} - \mu_k)(\mathbf{x}^{(i)} - \mu_k)^T}{\sum_{i=1}^{n} q_{i,k}}$$

$$\phi_j = \frac{1}{n}\sum_{i=1}^{n} Q^{(i)}(\mathbf{z}^{(j)})$$

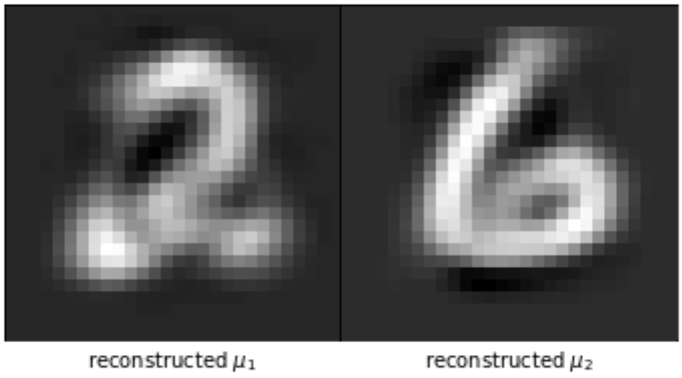
c. Implement EM algorithm yourself. Use the following initialization•initialization for mean: random Gaussian vector with zero mean•initialization for covariance: generate two Gaussian random matrix of size n-by-n :S_1 and S_2, and initialize the covariance matrix for the two components are $\Sigma_1=S_1S_{T1}+I_n$, and $\Sigma_2=S_2S_{T2}+I_n$, where I_n is an identity matrix of size n-by-n.Plot the log-likelihood function versus the number of iterations to show your algorithm is converging.

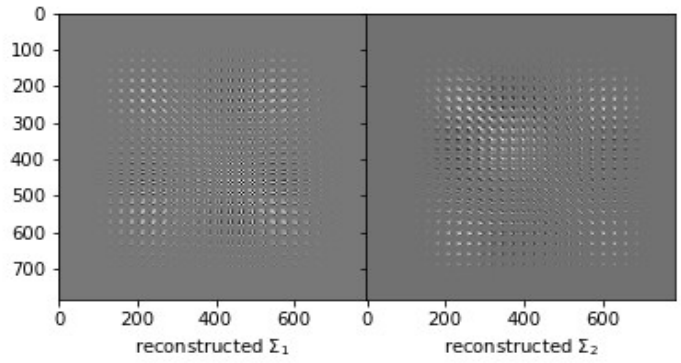


d. Report, the fitted GMM model when EM has terminated in your algorithms as follows. Make sure to report the weights for each component, and the mean of each component (you can reformat the vector to make them into 28-by-28 matrices and show images). Ideally, you should be able to see these means corresponds to “average” images. Report the two 784-by-784 covariance matrices by visualize their intensities.

component weights: [0.49313429 0.50686571]

component means:
[[-1.20409102 -4.41784437 -0.27699913 4.96306035 0.87272481]
 [2.89673174 -3.27123181 -0.05445827 4.93065369 0.49325342]]





e. Use the τ_{ik} to infer the labels of the images, and compare with the true labels. Report the misclassification rate for digits “2” and “6” respectively. Perform K-means clustering with K= 2 (you may call a package or use the code from your previous homework). Find out the misclassification rate for digits “2” and “6” respectively, and compare with GMM. Which one achieves the better performance?

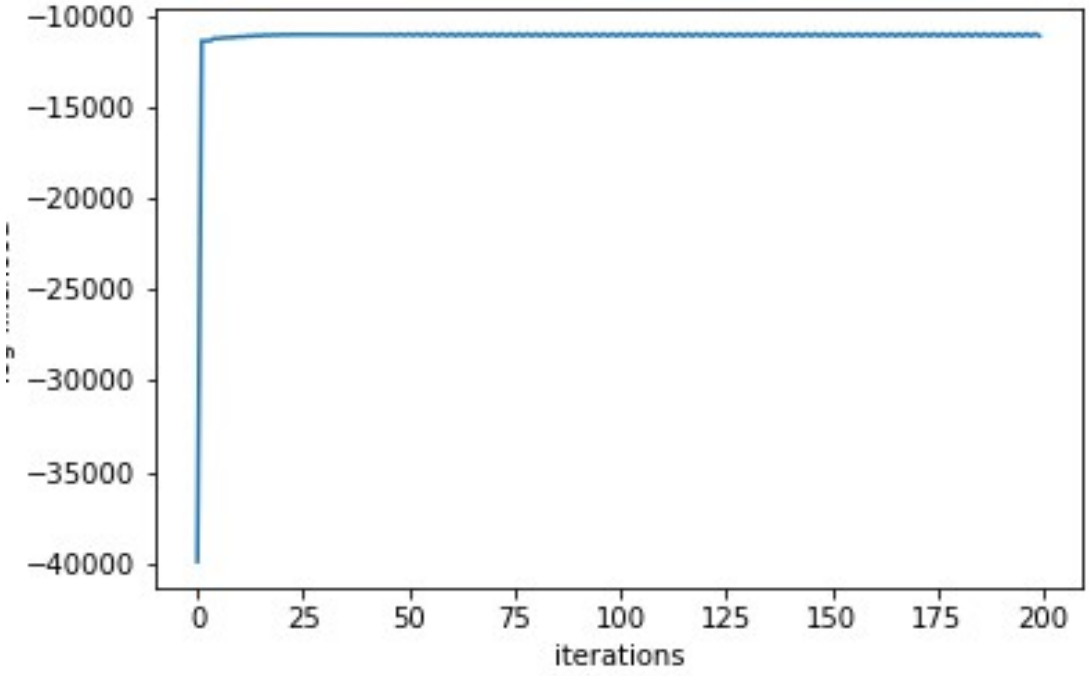
GMM vs K-Mean Misclassfication Rate

	GMM	K-Means
2 Misclassification Rate	5.81%	7.36%
6 Misclassification Rate	0.73%	4.70%
Total Misclassification Rate	3.37%	6.08%

GMM achieves better performance.

3. Implementing EM for MNIST dataset, with low-rank approximation.

a. Implement the low-rank approximation (with r= 5) and perform EM algorithm. Use the similar initialization method as the last question. Plot the log-likelihood function versus the number of iterations to show your algorithm is converging.



b. Use the τ_{ik} to infer the labels of the images, and compare with the true labels. Report the misclassification rate for digits “2” and “6” respectively. Compare with GMM using PCA that you have implemented in Question 2. Which one achieves better performance?

	GMM	GMM Low Rank Approx r=5
2 Misclassification Rate	5.81%	32.66%
6 Misclassification Rate	0.73%	35.49%
Total Misclassification Rate	3.37%	34.02%

With an r=5 GMM using PCA achieves better performance. Performance greatly improves for GMM using low rank approximation with higher r. It is also more consistent across both clusters.

GMM Low Rank Approximation Misclassification Rates for Different R values

	r=5	r=25	r=50
2 Misclassification Rate	32.66%	10.17%	4.46%
6 Misclassification Rate	35.49%	10.85%	4.49%
Total Misclassification Rate	34.02%	10.50%	4.47%