ISYE 6740, Spring 2024, Homework 3

100 points + 10 bonus points

Yuxi Chen

1. Conceptual questions. [20 points]

1. (5 points) Please compare the pros and cons of KDE over histogram, and give at least one advantage and disadvantage to each.

   The pro of using histogram is that it is easy to interpret and computationally more efficient than KDE. The con of using histogram is that it is not sample efficient for high dimensional data and outputs depend on bin size, which can cause noise and incorrect interpretation of data.
   The pro of using KDE is that it has better performance guarantees compared to histogram. KDE also allows the flexibility of adjusting bandwidth and kernel functions which is more suited for the given data, whereas histogram bins are fixed The con of using KDE is the high computation cost because of the way it stores the data in memory. It increases by mn where n is the sample size and may not be efficient if the sample size is large.

2. (5 points) Why you cannot use maximum likelihood estimation to directly estimate GMM? Then how to estimate the model of GMM?

   Typically to fit the model to the data, we will use maximum likelihood learning. We will need to solve for $\theta^* = argmaxl(\theta; D)$, which requires us to solve for the log-likelihood function $l(\theta; D)$. The reason we can not use maximum likelihood estimation directly is because we do not know the latent factors z^i, therefore we cannot evaluate $l(\theta; D)$ directly.
   So, as an alternative method to estimate the model of GMM, we will need to apply the EM algorithm. By using the EM algorithm, we can find the values of the latent values using Bayes rules in the E-step. Next in the M-step, we update the parameter values based on the latent values found in the E-step. We then repeat these steps until the parameters converge.

3. (5 points) For the EM algorithm for GMM, please show how to use the Bayes rule to drive τ_k^i in a closed-form expression.

   Starting with Bayes Rule, we have $P(z|x) = \frac{P(x|z)P(z)}{P(x)} = \frac{P(x,z)}{\sum_{z'} P(x,z')}$.
   Prior: $p(z) = \pi_z$

Likelihood: $p(x|z) = N(x|\mu_z, \Sigma_z)$
Normalization Constant: $p(x) = \sum_{z'} \pi_{z'} \mathcal{N}(x|\mu_{z'}, \Sigma_{z'})$

Now we are in the E-step to find the posterior distribution. Where for each data point x^i, we need to find $p(z^i = k|x^i)$ for each k.

$\tau_k^i = p(z_i = k|x, \theta^t) = \frac{p(x^i|z^i=k)p(z^i=k)}{\sum_{k'=1...K} p(z^i=k', x^i)}$

Which gives us, $\Rightarrow \frac{\pi_k \mathcal{N}(x^i|\mu_k, \Sigma_k)}{\sum_{k'=1...K} \pi_{k'} \mathcal{N}(x^i|\mu_{k'}, \Sigma_{k'})}$

4. (5 points) Explain how to choose the kernel bandwidth for KDE?

   Two common methods in selecting the best kernel bandwidth are Silverman's rule of thumb and cross validation.
   Silverman's rule of thumb lets us use $h = 1.06\hat{\sigma}m^{-1/5}$, where $\hat{\sigma}$ is the standard deviation of the samples. The result of the formula returns h, which is the kernel bandwidth. Silverman's rule can be applied to various kernel function as an initial estimate for bandwidth, but works well with Gaussian Kernels. When applied to other kernels, it will may additional adjustments.
   Cross validation is a better approach to find kernel bandwidth, but is more computationally expensive. In using cross validation we will randomly split the data into two sets and then obtain the kernel density estimation using one set. This is followed by, measuring the likelihood of the second set and we proceed to repeat this process and average the results. When used in comparison with the maximum likelihood, the point at which the curve is at the peak determines the optimum bandwidth.

2. Density estimation: Psychological experiments. [40 points]

In *Kanai, R., Feilden, T., Firth, C. and Rees, G., 2011. Political orientations are correlated with brain structure in young adults. Current biology, 21(8), pp.677-680.*, data are collected to study whether or not the two brain regions are likely to be independent of each other and considering different types of political view **For this question; you can use third party histogram and KDE packages; no need to write your own.** The data set n90pol.csv contains information on 90 university students who participated in a psychological experiment designed to look for relationships between the size of different regions of the brain and political views. The variables amygdala and acc indicate the volume of two particular brain regions known to be involved in emotions and decision-making, the amygdala and the anterior cingulate cortex; more exactly, these are residuals from the predicted volume, after adjusting for height, sex, and similar body-type variables. The variable orientation gives the students' locations on a five-point scale from 1 (very conservative) to 5 (very liberal). Note that in the dataset, we only have observations for orientation from 2 to 5.

Recall in this case, the kernel density estimator (KDE) for a density is given by

$$p(x) = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{h}K\left(\frac{x^i - x}{h}\right),$$

where x^i are two-dimensional vectors, $h > 0$ is the kernel bandwidth, based on the criterion we discussed in lecture. For one-dimensional KDE, use a one-dimensional Gaussian kernel

$$K(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}.$$

For two-dimensional KDE, use a two-dimensional Gaussian kernel: for

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2,$$

where x_1 and x_2 are the two dimensions respectively

$$K(x) = \frac{1}{2\pi}e^{-\frac{(x_1)^2+(x_2)^2}{2}}.$$

(a) (5 points) 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. Decide on a suitable number of bins so you can see the shape of the distribution clearly. Set an appropriate kernel bandwidth $h > 0$.

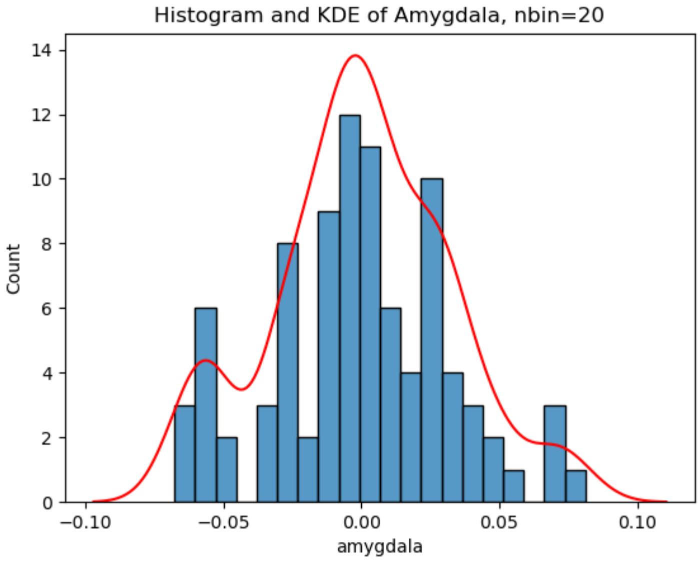


Figure 1: Histogram and KDE of Amygdala, nbin=20

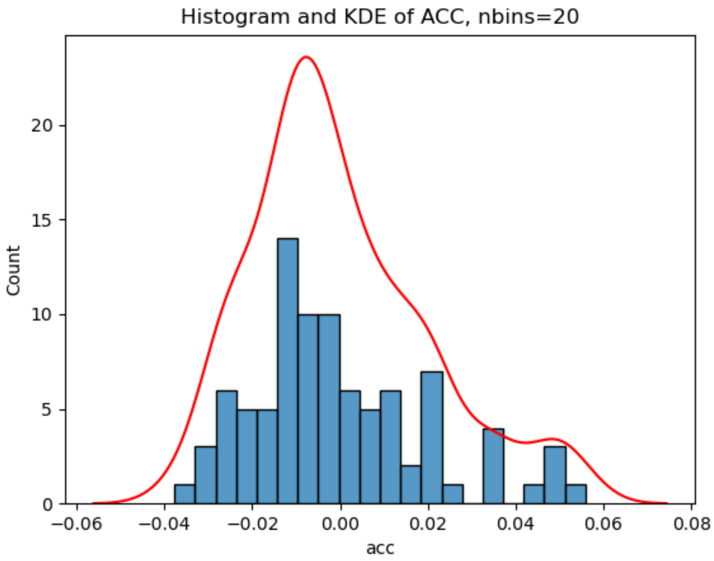


Figure 2: Histogram and KDE of ACC, nbin=20

(b) (5 points) 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.

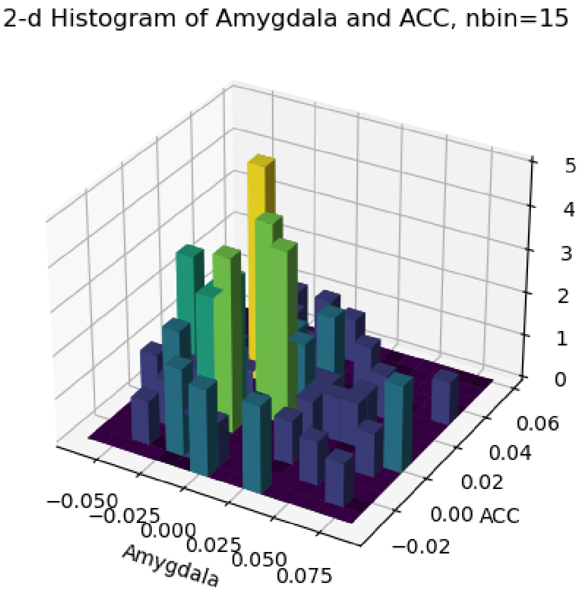


Figure 3: 2-d Histogram of Amygdala and ACC, nbin=15

(c) (15 points) Use kernel-density-estimation (KDE) to estimate the 2-dimensional density function of (amygdala, acc) (this means for this question, you can ignore the variable orientation). Set an appropriate kernel bandwidth $h > 0$.

Please show the two-dimensional KDE (e.g., two-dimensional heat-map, two-dimensional contour plot, etc.)

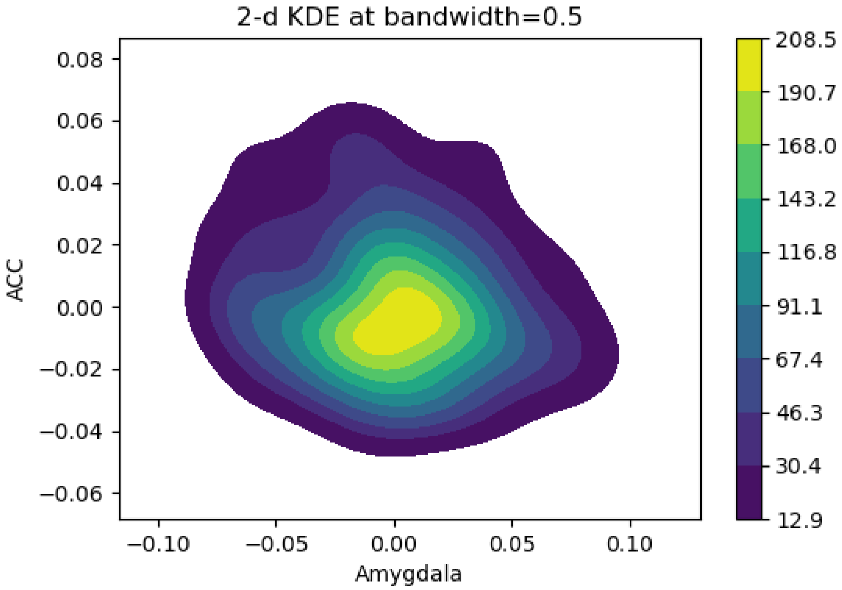
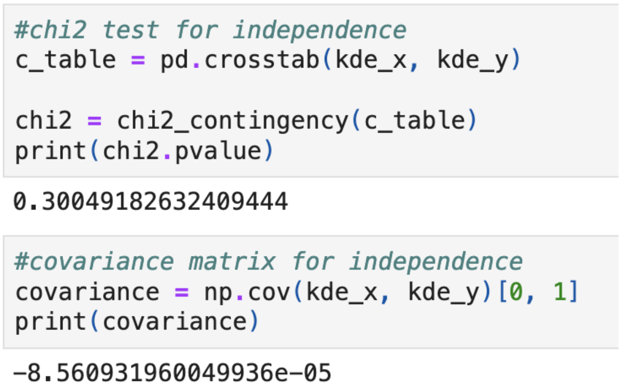


Figure 4: 2-d KDE plot at bandwidth 0.5

Please explain what you have observed: is the distribution unimodal or bi-modal? Are there any outliers?

From the figure above, we can see that the distribution is unimodal as there is only area of peak density. No, no outliers can be observed from this plot.

Are the two variables (amygdala, acc) likely to be independent or not? Please support your argument with reasonable investigations.



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

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

```
0.30049182632409444
```

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

```
-8.560931960049936e-05
```

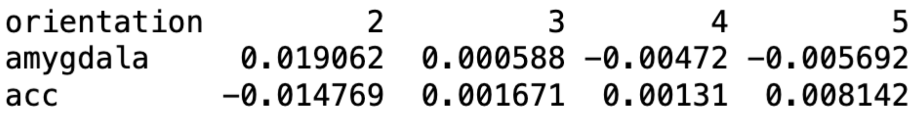
Figure 5: Tests for independence

From the chi sq test, we obtained the p-value of 0.3, which is greater than 0.05 indicating the two variables are indepedent of each other. Additionally, we performed a covariance test and the results obtained are near zero indicating non-linearity, which can support independence. Lastly, if we look at the 2-d KDE plot, we can see that the density appears to be mostly uniform and lack of patterns/clustering.

(d) (10 points) We will consider the variable orientation and consider conditional distributions. Please plot the estimated conditional distribution of amygdala conditioning on political orientation: $p(\text{amygdala}|\text{orientation} = c)$, $c = 2, \ldots, 5$, using KDE. Set an appropriate kernel bandwidth $h > 0$. Do the same for the volume of the acc: plot $p(\text{acc}|\text{orientation} = c)$, $c = 2, \ldots, 5$ using KDE. (Note that the conditional distribution can be understood as fitting a distribution for the data with the same orientation. Thus you should plot 8 one-dimensional distribution functions in total for this question.)

Now please explain based on the results, can you infer that the conditional distribution of amygdala and acc, respectively, are different from $c = 2, \ldots, 5$? This is a type of scientific question one could infer from the data: Whether or not there is a difference between brain structure and political view.

Now please also fill out the *conditional sample mean* for the two variables:



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

Figure 6: Conditional Sample Means

Remark: As you can see this exercise, you can extract so much more information from density estimation than simple summary statistics (e.g., the sample mean) in terms of explorable data analysis.

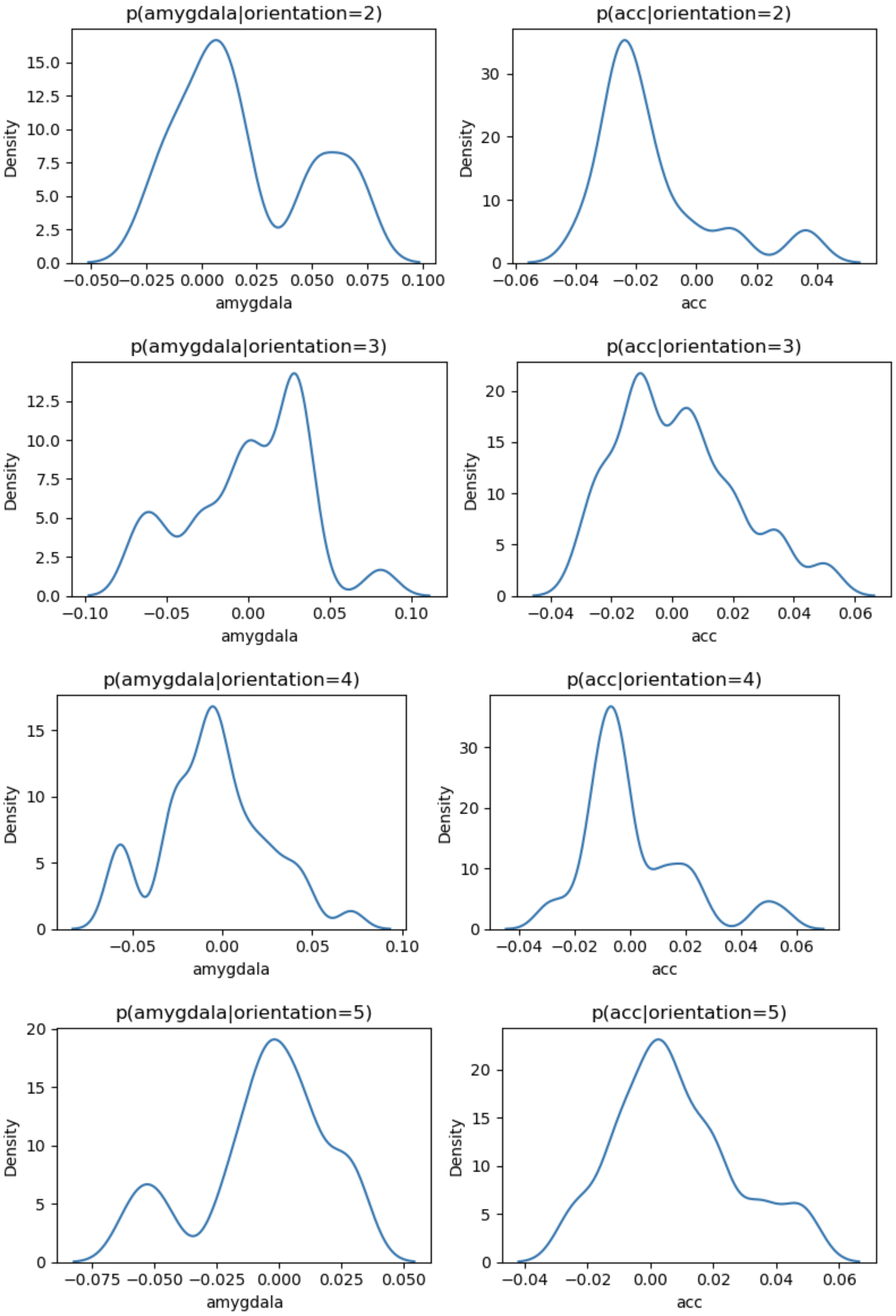


Figure 7: Conditional distribution on Political Orientation

From the conditional sample mean values we can see that the values for the amydala decreases as c increases from 2 to 5. Whereas, acc increases as we go from c=2 to 3, but then decreases again. From this, we can infer that there is a possibility of correlation between brain structure and political view, however, we will need to conduct further analysis to prove so.

(e) (5 points) 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: $p(\mathsf{amygdala}, \mathsf{acc}|\mathsf{orientation} = c)$, $c = 2, \ldots, 5$. You will use two-dimensional KDE to achieve the goal; et an appropriate kernel bandwidth $h > 0$. Please show the two-dimensional KDE (e.g., two-dimensional heat-map, two-dimensional contour plot, etc.).

Please explain based on the results, can you infer that the conditional distribution of two variables (amygdala, acc) are different from $c = 2, \ldots, 5$? This is a type of scientific question one could infer from the data: Whether or not there is a difference between brain structure and political view.

From the conditional joint distribution plots below, we can see that the highest density areas shift from the left to right as we increase the orientation from 2 to 5. This indicates that there may be a different in brain structure and political view.

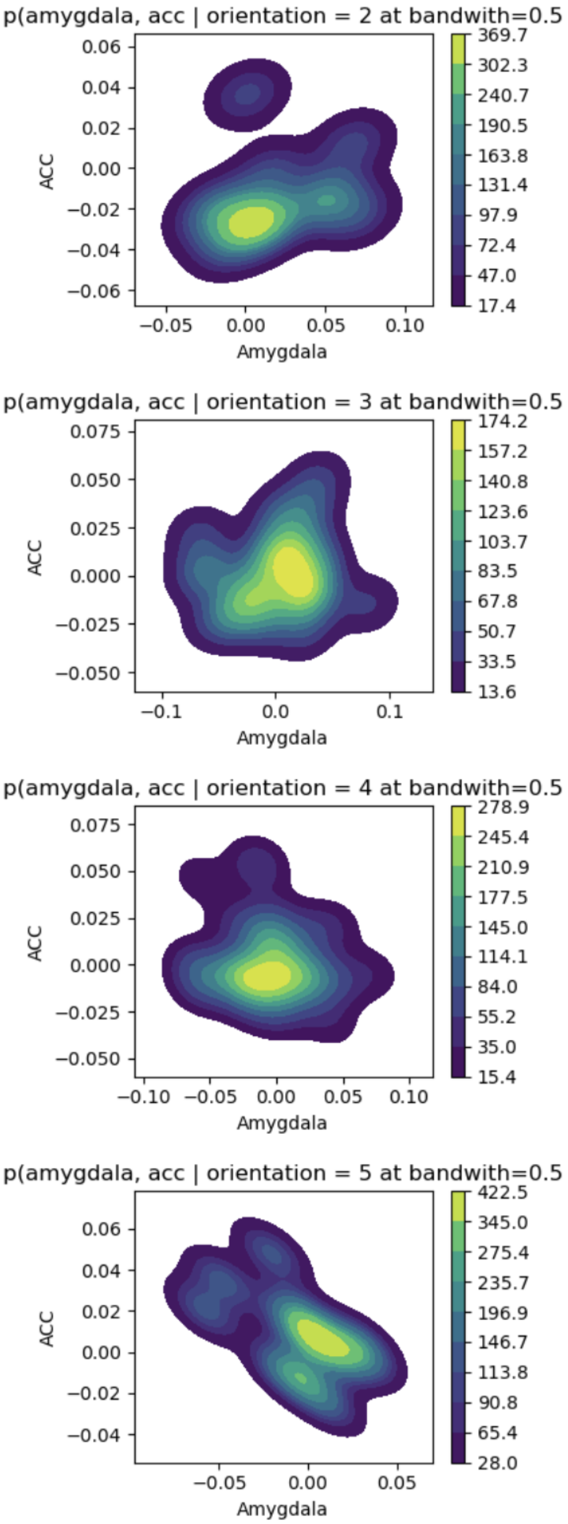


Figure 8: Joint distribution on Political Orientation

3. Implementing EM for MNIST dataset. [40 points]

Implement the EM algorithm for fitting a Gaussian mixture model for the MNIST handwritten digits dataset. For this question, we reduce the dataset to be only two cases, of digits "2" and "6" only. Thus, you will fit GMM with $C = 2$. Use the data file data.mat or data.dat. True label of the data are also provided in label.mat and label.dat.

The matrix images is of size 784-by-1990, i.e., there are 1990 images in total, and each column of the matrix corresponds to one image of size 28-by-28 pixels (the image is vectorized; the original image can be recovered by mapping the vector into a matrix).

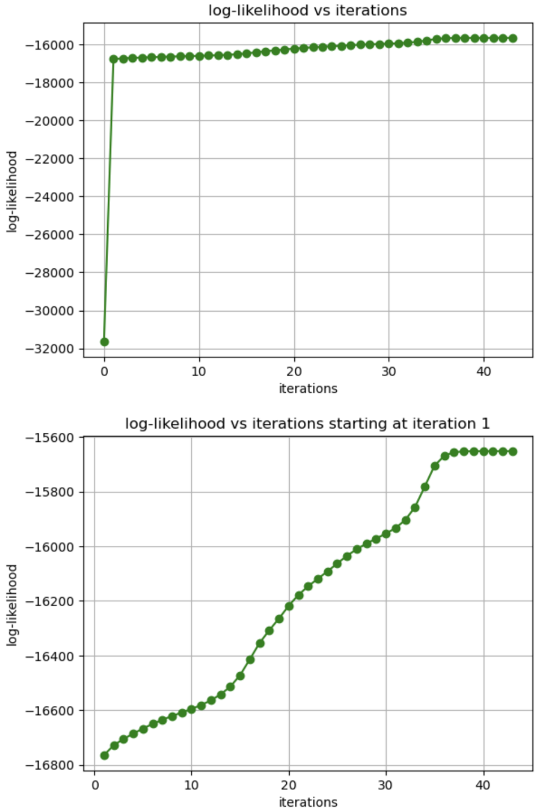
First use PCA to reduce the dimensionality of the data before applying to EM. We will put all "6" and "2" digits together, to project the original data into 4-dimensional vectors.

Now implement EM algorithm for the projected data (with 4-dimensions).
(In this question, we use the same set of data from the provided data files for training and testing)

(a) (10 points) 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 $\Sigma_1 = S_1 S_1^T + I_n$, and $\Sigma_2 = S_2 S_2^T + I_n$, where I_n is an identity matrix of size n-by-n.

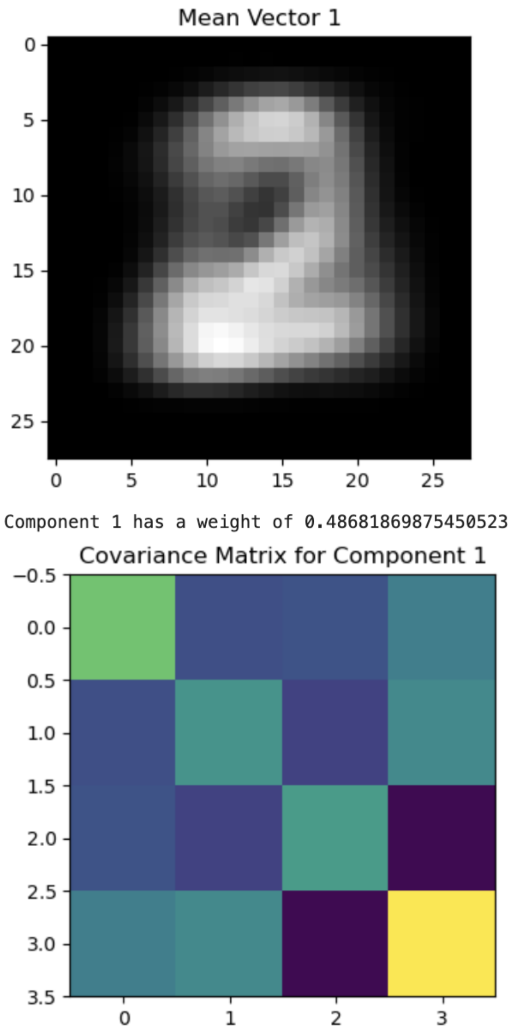
Plot the log-likelihood function versus the number of iterations to show your algorithm is converging.



```
Converges in 43 iterations
```

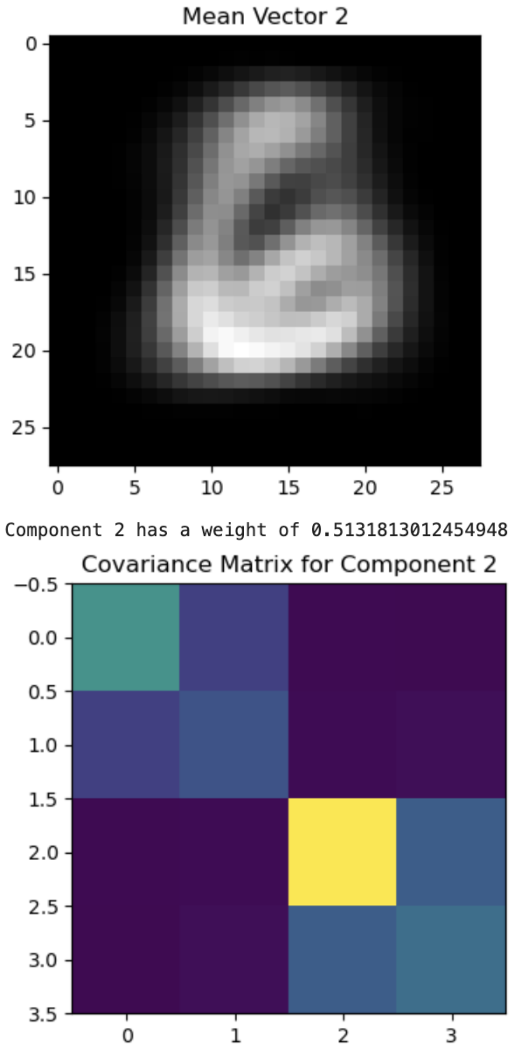
Figure 9: log-likelihood vs iterations.

(b) (20 points) Report, the fitted GMM model when EM has terminated in your algorithms as follows. Report the weights for each component, and the mean of each component, by mapping them back to the original space and 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 some kind of "average" images. You can report the two 4-by-4 covariance matrices by visualizing their intensities (e.g., using a gray scaled image or heat map).



```
Component 1 has a weight of 0.48681869875450523
```

Figure 10: mean vector 1

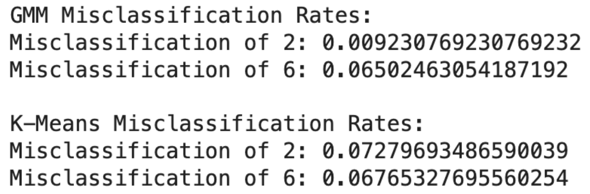


```
Component 2 has a weight of 0.5131813012454948
```

Figure 11: mean vector 2

(c) (10 points) Use the τ_k^i to infer the labels of the images, and compare with the true labels. Report the mis-classification 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 mis-classification rate for digits "2" and "6" respectively, and compare with GMM. Which one achieves the better performance?

In Figure 4, we have the misclassification rates using GMM and K-means. We can see that the misclasification of "6" using both GMM and KMeans are produced at almost the same rate. Whereas, the misclassification of "2" is much lower using GMM at less than one percent.



```
GMM Misclassification Rates:
Misclassification of 2: 0.009230769230769232
Misclassification of 6: 0.06502463054187192

K–Means Misclassification Rates:
Misclassification of 2: 0.07279693486590039
Misclassification of 6: 0.06765327695560254
```

Figure 12: Misclassification rate for "2" and "6"

4. De-bias review system using EM. [Bonus, 10 points]

In this question, we will develop an algorithm to remove individual reviewer's bias from their score. Consider the following problem. There are P papers submitted to a machine learning conference. Each of R reviewers reads each paper, and gives it a score indicating how good he/she thought that paper was. We let $x^{(pr)}$ denote the score that reviewer r gave to paper p. A high score means the reviewer liked the paper, and represents a recommendation from that reviewer that it be accepted for the conference. A low score means the reviewer did not like the paper.

We imagine that each paper has some "intrinsic" true value that we denote by μ_p, where a large value means it's a good paper. Each reviewer is trying to estimate, based on reading the paper, what μ_p is; the score reported $x^{(pr)}$ is then reviewer r's guess of μ_p.

However, some reviewers are just generally inclined to think all papers are good and tend to give all papers high scores; other reviewers may be particularly nasty and tend to give low scores to everything. (Similarly, different reviewers may have different amounts of variance in the way they review papers, making some reviewers more consistent/reliable than others.) We let ν_r denote the "bias" of reviewer r. A reviewer with bias ν_r is one whose scores generally tend to be ν_r higher than they should be.

All sorts of different random factors influence the reviewing process, and hence we will use a model that incorporates several sources of noise. Specifically, we assume that reviewers's scores are generated by a random process given as follows:

$$y^{(p)} \sim \mathcal{N}(\mu_p, \sigma_p^2)$$

$$z^{(r)} \sim \mathcal{N}(\nu_r, \tau_r^2)$$

$$x^{(pr)} | y^{(p)}, z^{(r)} \sim \mathcal{N}(y^{(p)} + z^{(r)}, \sigma^2).$$

The variables $y^{(p)}$ and $z^{(r)}$ are independent; the variables (x, y, z) for different paper-reviewer pairs are also jointly independent. Also, we only ever observe the $x^{(pr)}$s; thus, the $y^{(p)}$s and $z^{(r)}$s are all latent random variables.

We would like to estimate the parameters μ_p, σ_p^2, ν_r, τ_r^2. If we obtain good estimates of the papers "intrinsic values" μ_p, these can then be used to make acceptance/rejection decisions for the conference.

We will estimate the parameters by maximizing the marginal likelihood of the data $\{x^{(pr)}; p = 1, \dots, P, r = 1, \dots, R\}$. This problem has latent variables $y^{(p)}$s and $z^{(r)}$s, and the maximum likelihood problem cannot be solved in closed form. So, we will use EM.

Your task is to derive the EM update equations. For simplicity, you need to treat only $\{\mu_p, \sigma_p^2; p = 1 \dots, P\}$ and $\{\nu_r, \tau_r^2; r = 1...R\}$ as parameters, i.e. treat σ^2 (the conditional variance of $x^{(pr)}$ given $y^{(p)}$ and $z^{(r)}$) as a fixed, known constant.

1. Derive the E-step (5 points)

   (a) The joint distribution $p(y^{(p)}, z^{(r)}, x^{(pr)})$ has the form of a multivariate Gaussian density. Find its associated mean vector and covariance matrix in terms of the parameters μ_p, σ_p^2, ν_r, τ_r^2 and σ^2. [Hint: Recognize that $x^{(pr)}$ can be written as $x^{(pr)} = y^{(p)} + z^{(r)} + \epsilon^{(pr)}$, where $\epsilon^{(pr)} \sim \mathcal{N}(0, \sigma^2)$ is independent Gaussian noise.

   (b) Derive an expression for $Q_{pr}(\theta'|\theta) = \mathbb{E}[\log p(y^{(p)}, z^{(r)}, x^{(pr)})|x^{(pr)}, \theta]$ using the conditional distribution $p(y^{(p)}, z^{(r)}|x^{(pr)})$ (E-step) (Hint, you may use the rules for conditioning on subsets of jointly Gaussian random variables.)

2. (5 points) Derive the M-step to update the parameters μ_p, σ_p^2, ν_r, and τ_r^2. [Hint: It may help to express an approximation to the likelihood in terms of an expectation with respect to $(y^{(p)}, z^{(r)})$ drawn from a distribution with density $Q_{pr}(y^{(p)}, z^{(r)})$.]

Remark: John Platt (whose SMO algorithm you've seen) implemented a method quite similar to this one to estimate the papers' true scores. (There, the problem was a bit more complicated because not all reviewers reviewed every paper, but the essential ideas are the same.) Because the model tried to estimate and correct for reviewers' biases, its estimates of the paper's value were significantly more useful for making accept/reject decisions than the reviewers' raw scores for a paper.