CS5785: Homework #4

Due on December 3, 2020

LATE SUBMISSION (submitted on December 6, 2020)

Tianyou Xiao (tx43), Ziyu Song (zs363)

1 Clustering for Text Analysis

We implemented K-Means by ourselves in this question. We initialized our K-Means function randomly.

1.1 (a)

We used a graph to show the relationship between value of k and sum of squared errors with k value from 1 to 20. Although when k is larger, the performance will be better; however, we chose to have $k = 5$ as our k value. From the graph below, we can find out that when $k = 5$, the performance is the most optimal after a large drop.

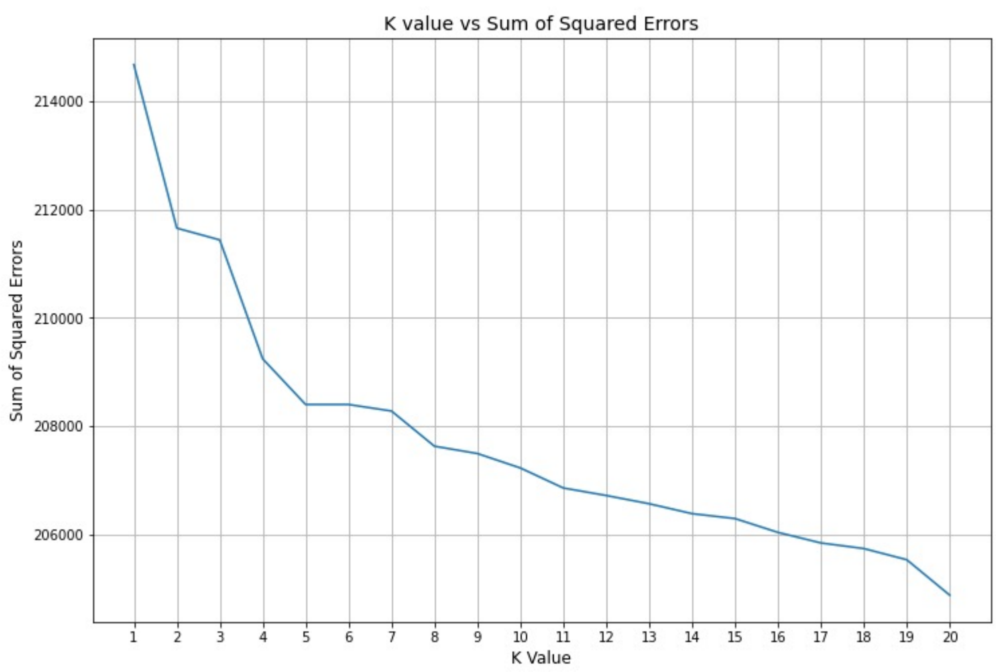

Figure 1: K Value vs. Sum of Squared Errors

Below are the top 10 words of each cluster:

```
i = 0
words =  ['retain', 'extraction', 'metals', 'oxidative', 'cns', 'nmr',
↪  'unrelated', 'sucrose', 'threefold', 'oscillation']
---------------------------------------------------
i = 1
words =  ['result', 'second', 'research', 'nature', 'addressed', 'science',
↪  'correspondence', 'end', 'references', 'contrast']
---------------------------------------------------
i = 2
```

```
words =  ['loading', 'soil', 'prior', 'regulates', 'consensus', 'offer',
↪  'investigation', 'meet', 'easy', 'explains']
--------------------------------------------------
i = 3
words =  ['protected', 'madison', 'evol', 'develop', 'candidates', 'patterning',
↪  'daily', 'bax', 'barrier', 'recycling']
--------------------------------------------------
i = 4
words =  ['computing', 'telescope', 'train', 'explaining', 'attraction',
↪  'oxford', 'interacting', 'radar', 'localized', 'permit']
--------------------------------------------------
```

Below are the top 10 documents of each cluster:

```
i = 0
Closest= [[ 62  32 235  45 121  70 278  47 203  97]]
Titles = [['Integrating the Many Aspects of Biology'
  'Designer Labs: Architecture Discovers Science'
  'Statistics of Ancestral Roots'
  'Influences of Dietary Uptake and Reactive Sulfides on Metal Bioavailability
  ↪  from Aquatic Sediments'
  'Corrections and Clarifications: A Short Fe-Fe Distance in Peroxodiferric
  ↪  Ferritin: Control of Fe Substrate versus Cofactor Decay?'
  'Late Cretaceous Polar Wander of the Pacific Plate: Evidence of a Rapid True
  ↪  Polar Wander Event'
  'Genetic Suppression of Polyglutamine Toxicity in Drosophila'
  'Communication through a Diffusive Medium: Coherence and Capacity'
  'Stem Cells Branch Out' 'HIV Infection and Dementia']]
--------------------------------------------------
i = 1
Closest= [[19 43 45 34 14 46 17 44 16 49]]
Titles = [['Generating Solitons by Phase Engineering of a Bose-Einstein
↪  Condensate'
  'Organizing Principles for a Diversity of GABAergic Interneurons and Synapses
  ↪  in the Neocortex'
  'Influences of Dietary Uptake and Reactive Sulfides on Metal Bioavailability
  ↪  from Aquatic Sediments'
  'Social Mentalizing Abilities in Mental Patients'
  'The Formation and Early Evolution of the Milky Way Galaxy'
  'Reaction of Plutonium Dioxide with Water: Formation and Properties of
  ↪  <latex>$PuO_{2+x}$</latex>'
  'The Galactic Center: An Interacting System of Unusual Sources'
  'Rutile-Bearing Refractory Eclogites: Missing Link between Continents and
  ↪  Depleted Mantle'
  'The Baryon Halo of the Milky Way: A Fossil Record of Its Formation'
```

```
  'Formation of Cyclic Water Hexamer in Liquid Helium: The Smallest Piece of
   ↪  Ice']]
--------------------------------------------------
i = 2
Closest= [[263 354 127   0  37 107 286 355 164 356]]
Titles = [['Genomics: Journey to the Center of Biology' 'Running the Red Light'
  'Regulating Export of ER Cargo'
  "Baedeker's Guide, or Just Plain 'Trouble'?"
  'The Green Revolution Strikes Gold'
  'Complementary Neural Mechanisms for Tracking Items in Human Working Memory'
  'Reforming the Patent System'
  'Giant Birefringent Optics in Multilayer Polymer Mirrors'
  'Function of PI3Kg in Thymocyte Development, T Cell Activation, and Neutrophil
   ↪  Migration'
  'Dynamic Variations at the Base of the Solar Convection Zone']]
--------------------------------------------------
i = 3
Closest= [[213 257 451  23  83  35 177 567 170 531]]
Titles = [['Why Stem Cells?'
  'Wildlife Deaths Are a Grim Wake-Up Call in Eastern Europe'
  "Working in the Hot Zone: Galveston's Microbe Hunters"
  'The Changing Morphology and Increasing Deceleration of Supernova 1993J in M81'
  'A Role for Histone Acetylation in the Developmental Regulation of V(D)J
   ↪  Recombination'
  'Corrections and Clarifications: Identification of a Mating Type-like Locus in
   ↪  the Asexual Pathogenic Yeast Candida albicans'
  'A Powerhouse Divided'
  'Chromatin-Independent Nuclear Envelope Assembly Induced by Ran GTPase in
   ↪  Xenopus Egg Extracts'
  'Ecologists on a Mission to Save the World'
  "Prometheus: Io's Wandering Plume"]]
--------------------------------------------------
i = 4
Closest= [[36 23 20 42  3 11 29 17 16 18]]
Titles = [['Waiting for Organ Transplantation'
  'The Changing Morphology and Increasing Deceleration of Supernova 1993J in M81'
  'Equilibrium Regained: From Nonequilibrium Chaos to Statistical Mechanics'
  'Cenozoic Deep-Sea Temperatures and Global Ice Volumes from Mg/Ca in Benthic
   ↪  Foraminiferal Calcite'
  'Corrections and Clarifications: Commercialization of Genetic Research and
   ↪  Public Policy'
  'A Crushing End for Our Galaxy' 'Tbx5 and the Retinotectum Projection'
  'The Galactic Center: An Interacting System of Unusual Sources'
```

```
  'The Baryon Halo of the Milky Way: A Fossil Record of Its Formation'
  'Structural Basis of Smad2 Recognition by the Smad Anchor for Receptor
  ↪  Activation']]
--------------------------------------------------
```

This algorithm is able to cluster documents with similar topic or based on similar words, such as documents with biology and chemistry topics, but we don't see clear patterns from the farthest words. Such an algorithm is useful when we want to search for literature by topic or tags rather than exact titles. Also, since the initial centroids of k-means is randomly selected from the dataset, the result will change in each run, and the tendency is sometimes hard to capture.

1.2 (b)

In this section, we chose k value equals to 6.

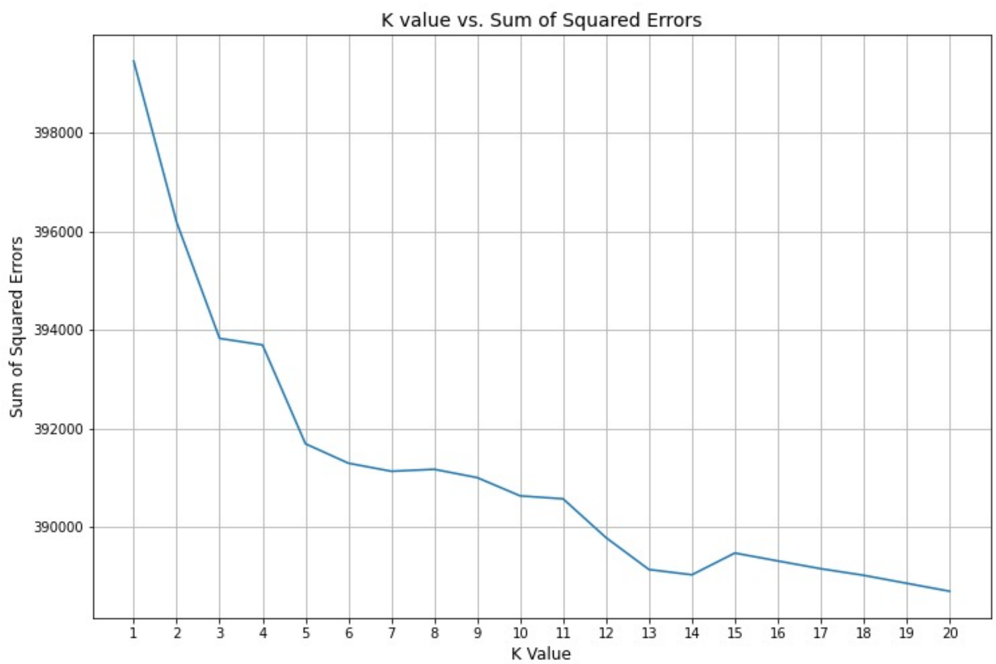

Figure 2: K Value vs. Sum of Squared Errors

Below are the top 10 words of each cluster:

```
i = 0
words =  ['get', 'challenge', 'ratio', 'nature', 'block', 'create', 'mars',
↪  'serum', 'phosphate', 'forms']
--------------------------------------------------
i = 1
words =  ['images', 'subunit', 'develop', 'determined', 'conventional', 'forest',
↪  'negative', 'recent', 'allows', 'regulation']
```

```
--------------------------------------------------
i = 2
words =  ['metal', 'normally', 'estimated', 'site', 'neuronal', 'reaction',
↪  'chemistry', 'systems', 'physical', 'orbital']
--------------------------------------------------
i = 3
words =  ['significant', 'environmental', 'top', 'web', 'rotation', 'diffusion',
↪  'vesicles', 'host', 'mitochondrial', 'evidence']
--------------------------------------------------
i = 4
words =  ['energy', 'survival', 'proliferation', 'biology', 'affinity', 'plus',
↪  'take', 'providing', 'extracts', 'immune']
--------------------------------------------------
i = 5
words =  ['resolution', 'late', 'gray', 'host', 'dimensional', 'mechanical',
↪  'strand', 'tumor', 'ca2', 'interaction']
--------------------------------------------------
```

Below are the top 10 documents of each cluster:

```
i = 0
Closest = [[ 4 31  1 10 29  2 21 19  8 11]]
Titles = [['Corrections and Clarifications: Commercialization of Genetic Research
↪  and Public Policy'
  'A Mammalian <latex>$H^+$</latex> Channel Generated through Alternative
  ↪  Splicing of the NADPH Oxidase Homolog NOH-1'
  "Baedeker's Guide, or Just Plain 'Trouble'?"
  "Charon's First Detailed Spectra Hold Many Surprises"
  'DNA Topoisomerase IIb and Neural Development'
  "Duchamp's L.H.O.O.Q.-From 1919 or 1930?"
  'Equilibrium Regained: From Nonequilibrium Chaos to Statistical Mechanics'
  'Structural Basis of Smad2 Recognition by the Smad Anchor for Receptor
  ↪  Activation'
  'Tracing the Origins of Salmonella Outbreaks'
  'Nota Bene: Sensing Old Age']]
--------------------------------------------------
i = 1
Closest = [[ 9  2 31 14  0  6  4 11  1  8]]
Titles = [['Reading the Worm Genome' "Duchamp's L.H.O.O.Q.-From 1919 or 1930?"
  'A Mammalian <latex>$H^+$</latex> Channel Generated through Alternative
  ↪  Splicing of the NADPH Oxidase Homolog NOH-1'
  'Into the Lair of the Beast' 'Archaeology in the Holy Land'
  'Will Tribal Knowledge Survive the Millennium?'
  'Corrections and Clarifications: Commercialization of Genetic Research and
  ↪  Public Policy'
```

```
  'Nota Bene: Sensing Old Age'
  "Baedeker's Guide, or Just Plain 'Trouble'?"
  'Tracing the Origins of Salmonella Outbreaks']]
--------------------------------------------------
i = 2
Closest = [[46 18 37 33 26  8 12 29  3  4]]
Titles = [['Influences of Dietary Uptake and Reactive Sulfides on Metal
↪  Bioavailability from Aquatic Sediments'
  'The Galactic Center: An Interacting System of Unusual Sources'
  'Waiting for Organ Transplantation'
  'Designer Labs: Architecture Discovers Science'
  'A Short Fe-Fe Distance in Peroxodiferric Ferritin: Control of Fe Substrate
   ↪  versus Cofactor Decay?'
  'Tracing the Origins of Salmonella Outbreaks'
  'A Crushing End for Our Galaxy'
  'DNA Topoisomerase IIb and Neural Development'
  'Resistance to Bt Toxins'
  'Corrections and Clarifications: Commercialization of Genetic Research and
   ↪  Public Policy']]
--------------------------------------------------
i = 3
Closest = [[36 42  4 63 19 17 20  9  8  3]]
Titles = [['Corrections and Clarifications: Identification of a Mating Type-like
↪  Locus in the Asexual Pathogenic Yeast Candida albicans'
  'Deconstructing the Science Wars by Reconstructing an Old Mold'
  'Corrections and Clarifications: Commercialization of Genetic Research and
   ↪  Public Policy'
  'Integrating the Many Aspects of Biology'
  'Structural Basis of Smad2 Recognition by the Smad Anchor for Receptor
   ↪  Activation'
  'The Baryon Halo of the Milky Way: A Fossil Record of Its Formation'
  'Generating Solitons by Phase Engineering of a Bose-Einstein Condensate'
  'Reading the Worm Genome' 'Tracing the Origins of Salmonella Outbreaks'
  'Resistance to Bt Toxins']]
--------------------------------------------------
i = 4
Closest = [[5 4 2 0 1 6 3]]
Titles = [['Corrections and Clarifications: First-Principles Determination of
↪  Elastic Anisotrophy and Wave Velocities of MgO at Lower Mantle Conditions'
  'Corrections and Clarifications: Commercialization of Genetic Research and
   ↪  Public Policy'
  "Duchamp's L.H.O.O.Q.-From 1919 or 1930?"
  'Archaeology in the Holy Land'
```

```
  "Baedeker's Guide, or Just Plain 'Trouble'?"
  'Will Tribal Knowledge Survive the Millennium?'
  'Resistance to Bt Toxins']]
--------------------------------------------------
i = 5
Closest = [[35 49 24 26 32  7 30 15  1  0]]
Titles = [['Social Mentalizing Abilities in Mental Patients'
  'Direct Observation of Dynamical Heterogeneities in Colloidal Hard-Sphere
  ↪  Suspensions'
  'The Changing Morphology and Increasing Deceleration of Supernova 1993J in M81'
  'A Short Fe-Fe Distance in Peroxodiferric Ferritin: Control of Fe Substrate
  ↪  versus Cofactor Decay?'
  'Stat3-Mediated Transformation of NIH-3T3 Cells by the Constitutively Active
  ↪  Q205L <latex>$G\\alpha_o$</latex> Protein'
  'Brane-Worlds' 'Tbx5 and the Retinotectum Projection'
  'The Formation and Early Evolution of the Milky Way Galaxy'
  "Baedeker's Guide, or Just Plain 'Trouble'?"
  'Archaeology in the Holy Land']]
--------------------------------------------------
```

This algorithm is more of a vocabulary clustering rather than document clustering. There is no clear topic for each cluster and it is also hard to find a clear topic from just the combination of words.

When comparing with document clustering and term clustering, it seems that document clustering is more helpful and easier to interpret. Terms normally cannot indicate the context of the literature directly. When grouping documents together, we are able to find out the topic from the grouping. Also, depending on how we cluster the features, the information retrieved from the grouping can be very different. Overall, using K-Means can help us to categorize titles and vocabularies based on analysis on articles efficiently and without manual observation. However, the performance really depends on the initialization. Bad initialization will lead to unmeaningful clustering with very few data points. The randomness of initialization makes the results changeable in each run.

2 EM Algorithm and Implementation

2.1 (a)

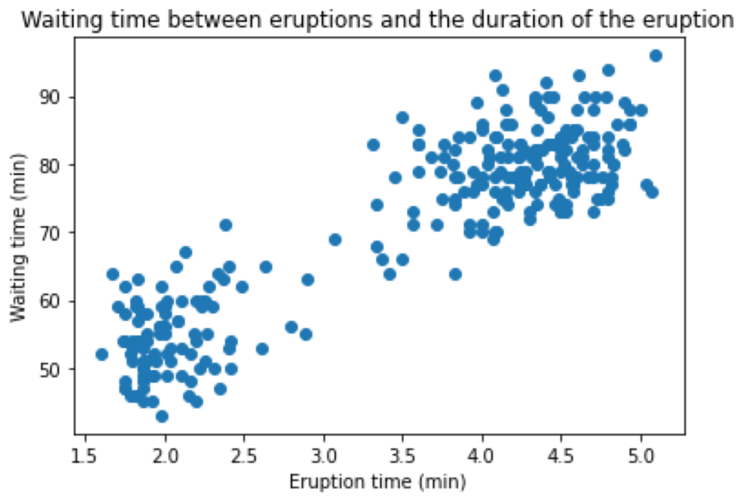

Figure 3: All Data Points on 2-D Plane

2.2 (b)

We initialize the μ to be 2 randomly selected data points from the dataset. The covariance matrix Σ is initialized to the variance of the whole dataset, and λ is randomly initialized to be a 2-dimension vector where the entries sum to 1.

Termination Criteria: When the change of μ of the models in the current iteration is smaller than 0.01, compared to the μ in the previous iteration, we terminate the algorithm. The reason we choose such criteria is that a very small change in the μ means that the model has already found the centers of each Gaussian model, and the algorithm converges.

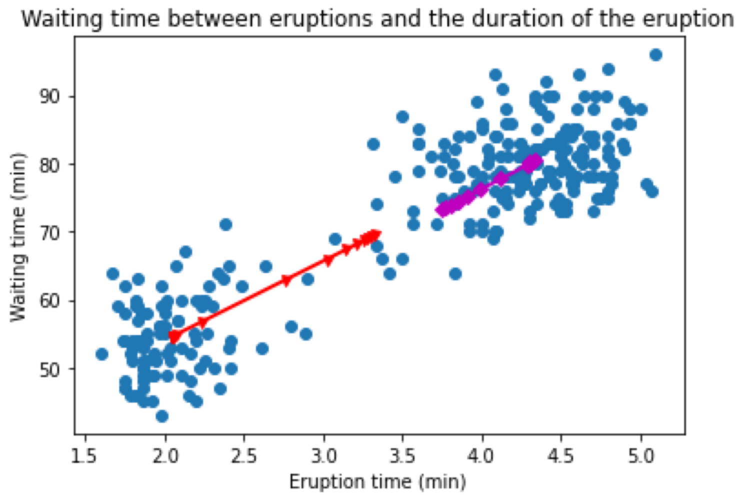

Figure 4: Trajectories of two mean vectors in 2 dimensions

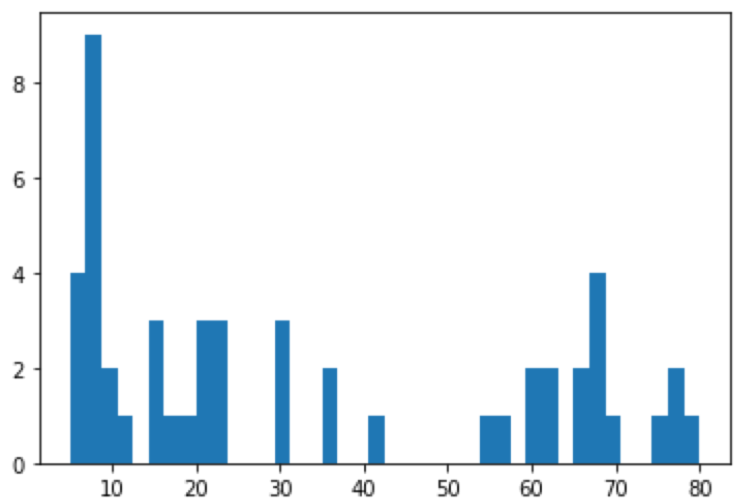

Figure 5: Run the program for 50 times

2.3 (c)

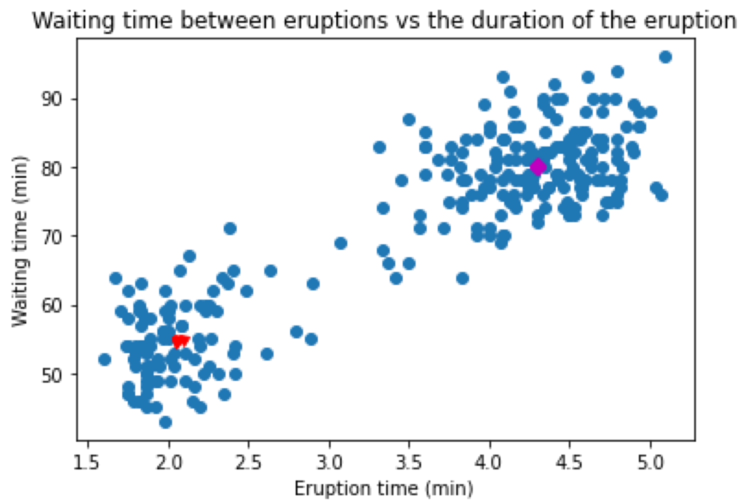

Figure 6: Run a k-means algorithm over all the data points with K = 2

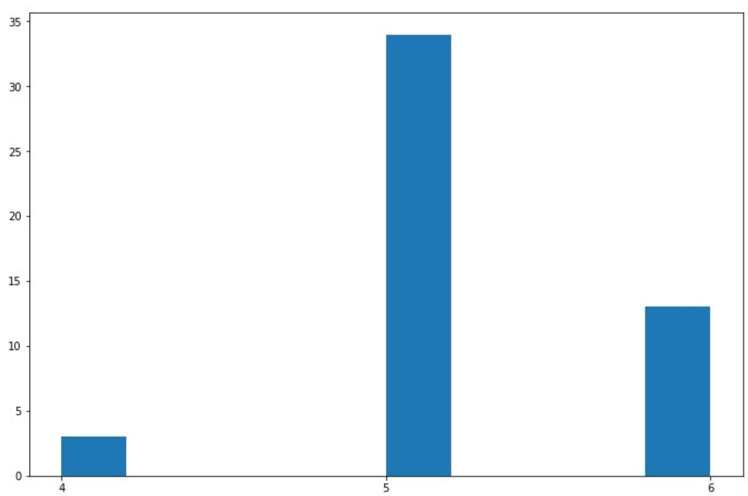

Figure 7: Distribution of Iteration

In problem (c) we follow the same termination criteria as in the previous question. The algorithm with random initialization converges much slower on average than the one with parameters generated by K-means and maximum likelihood estimation. Also, the algorithm in problem (b) may get stuck in local minima, depending on the choice of initial mu, while the (c) algorithm always converge to the global maxima. We believe that this is reasonable, since one big problem of GMM is that it's possible to get stuck at local minima, and providing some "background" information to the model will largely resolve such problem.

3 Eigenface for Face Recognition

For this question, we implemented the eigendecomposition for feature reduction, and we created a logistics regression prediction model based on rank-r approximation and recognized images of human faces using the prediction model.

3.1 (a)

Downloaded The Face Dataset. Imported and reshaped the training and testing dataset.

3.2 (b)

Pick a face image from $\mathbf{X}$ and display the image in grayscale for both training set and testing set.

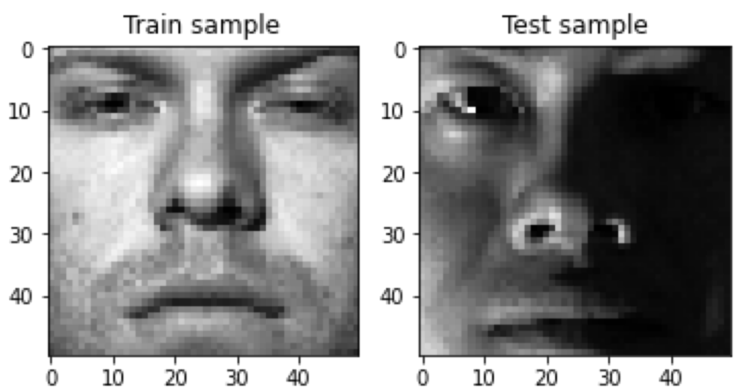

Figure 8: Image Displayed in Grayscale

3.3 (c) Average Face

Computed the average face μ from the training set by taking the mean of each column of every sample. We then displayed the average face below:

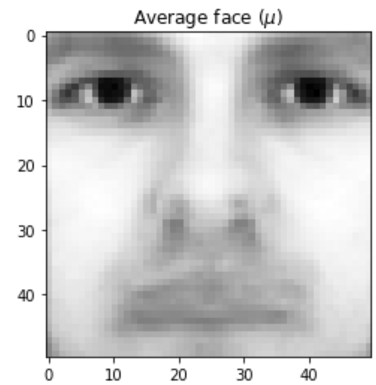

Figure 9: Average Face from the Whole Training Set

3.4 (d) Mean Subtraction

We subtracted the average face vector from every row in the training data by subtracting μ from every column in the set. We then displayed a face image in grayscale after mean subtraction for both the training set and testing set.

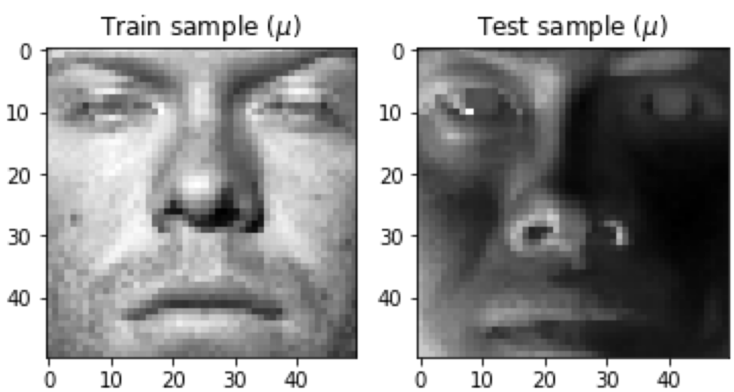

Figure 10: A Face Image After Mean Subtraction

3.5 (e) Eigenface

We performed Singular Value Decomposition (SVD) on the training set. The training set got split up into vectors $\mathbf{U}$, $\mathbf{\Sigma}$, and $\mathbf{V}^T$, and we implemented the `numpy.linalg.svd` function to do this. We then displayed the first 10 eigenfaces in grayscale.

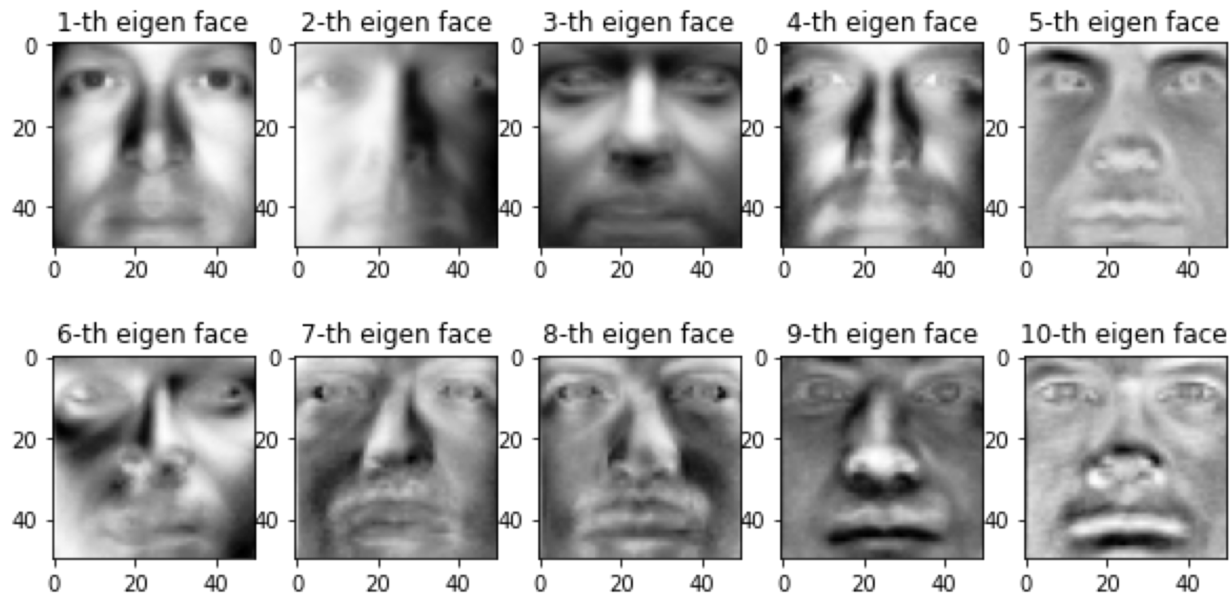

Figure 11: The First 10 Eigenfaces in Grayscale

3.6 (f) Eigenface Feature

We computed the dot product of training data with the transpose of first r elements of $\mathbf{V}$ transpose.

Here is the function we implemented to generate r-dimensional feature matrix $\mathbf{F}$ and $\mathbf{F}_{\mathbf{test}}$ for training images $\mathbf{X}$ and testing images $\mathbf{X}_{\mathbf{test}}$.

```
def eigenface_feature(V, train, test, r):
    F = np.dot(train, V[:r,:].T)
    F_test = np.dot(test, V[:r,:].T)
    return (F, F_test)
```

3.7 (g) Face Recognition

We used a for loop to compute the classification accuracy on test set for r from 1 to 200. We used `LogisticRegression` and `OneVsRestClassifier` imported from the `sklearn` library.

Here is a plot showing the accuracy score on the test set:

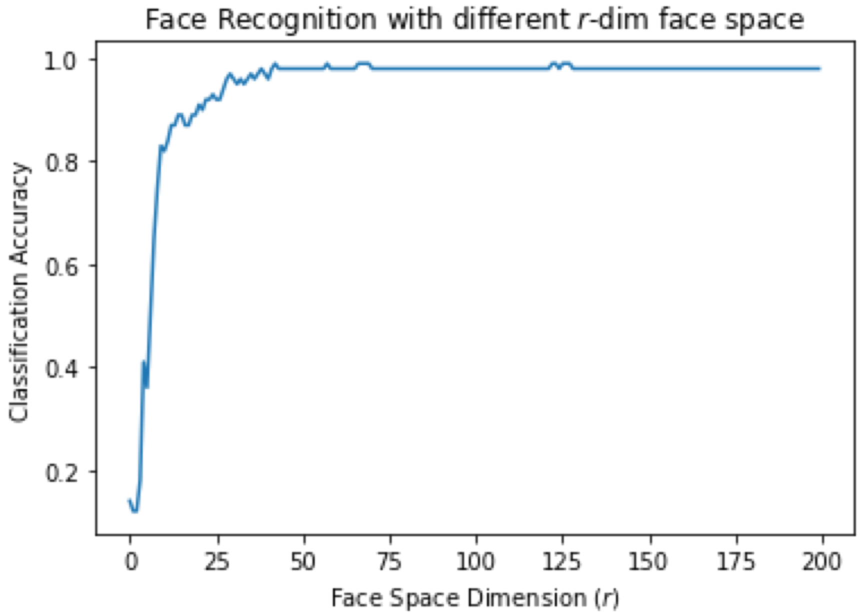

Figure 12: Face Recognition with Different r-dimension Face Space

4 Written Exercises

4.1 SVD and Eigendecomposition

Given the Singular Value Decomposition (SVD) of a matrix $\mathbf{X}$: $\boldsymbol{X} = U\Sigma V^{\mathrm{T}}$.
Given that $\boldsymbol{U}$ is an unitary matrix, we have: $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \boldsymbol{I}$. Then we bring this to the following equation:

$$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} = \left(U\Sigma V^{\mathrm{T}}\right)^{\mathrm{T}}\left(U\Sigma V^{\mathrm{T}}\right) = V\Sigma^{\mathrm{T}}U^{\mathrm{T}}U\Sigma V^{\mathrm{T}} = \boldsymbol{V}\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}\boldsymbol{V}^{\mathrm{T}}$$

On the right, we have the eigenvector matrix $\mathbf{V}$ and the eigenvalue matrix $\boldsymbol{\Sigma}^{\mathrm{T}}\boldsymbol{\Sigma}$.

4.2 Weights for Clustering

$$\begin{aligned}
d_e^{(w)}\left(x_i, x_{i'}\right) &= \frac{\sum_{l=1}^{p} w_l\left(x_{il} - x_{i'l}\right)^2}{\sum_{l=1}^{p} w_l} \\
&= \frac{\sum_{l=1}^{p}\left[\sqrt{w_l}\left(x_{il} - x_{i'l}\right)\right]^2}{\sum_{l=1}^{p} w_l} \\
&= \sum_{l=1}^{p}\left[\sqrt{\frac{w_l}{\sum w_l}}\left(x_{il} - x_{i'l}\right)\right]^2
\end{aligned}$$

Given that

$$z_{il} = x_{il} \cdot \left(\frac{w_l}{\sum_{l=1}^{p} w_l} \right)^{1/2}$$

We could have

$$\begin{aligned} d_e^{(w)}\left(x_i, x_{i'}\right) &= \sum_{l=1}^{p} \left(z_{il} - z_{i'l}\right)^2 \\ &= d_e\left(z_i, z_{i'}\right) \end{aligned}$$

4.3 SVD of Rank Deficient Matrix

4.3.1 (a) Compute

$$\boldsymbol{M}^{\mathrm{T}}\boldsymbol{M} = \begin{bmatrix} 1 & 3 & 2 & 0 & 5 \\ 0 & 7 & -2 & -1 & 8 \\ 3 & 2 & 8 & 1 & 7 \end{bmatrix} \begin{bmatrix} 1 & 0 & 3 \\ 3 & 7 & 2 \\ 2 & -2 & 8 \\ 0 & -1 & 1 \\ 5 & 8 & 7 \end{bmatrix} = \begin{bmatrix} 39 & 57 & 60 \\ 57 & 118 & 53 \\ 60 & 53 & 127 \end{bmatrix}$$

$$\boldsymbol{M}\boldsymbol{M}^{\mathrm{T}} = \begin{bmatrix} 1 & 0 & 3 \\ 3 & 7 & 2 \\ 2 & -2 & 8 \\ 0 & -1 & 1 \\ 5 & 8 & 7 \end{bmatrix} \begin{bmatrix} 1 & 3 & 2 & 0 & 5 \\ 0 & 7 & -2 & -1 & 8 \\ 3 & 2 & 8 & 1 & 7 \end{bmatrix} = \begin{bmatrix} 10 & 9 & 26 & 3 & 26 \\ 9 & 62 & 8 & -5 & 85 \\ 26 & 8 & 72 & 10 & 50 \\ 3 & -5 & 10 & 2 & -1 \\ 26 & 85 & 50 & -1 & 138 \end{bmatrix}$$

4.3.2 (b) eigenvalues

For $\mathbf{M}^T\mathbf{M}$:

$$\begin{vmatrix} 39-\lambda & 57 & 60 \\ 57 & 118-\lambda & 53 \\ 60 & 53 & 127-\lambda \end{vmatrix} = -\lambda^3 + 284\lambda^2 - 14883\lambda = 0$$

$$\lambda_1 = 69.3295$$

$$\lambda_2 = 214.6705$$

$$\lambda_3 = 0$$

For $\mathbf{M}\mathbf{M}^T$:

$$\begin{vmatrix} 10-\lambda & 9 & 26 & 3 & 26 \\ 9 & 62-\lambda & 8 & -5 & 85 \\ 26 & 8 & 72-\lambda & 10 & 50 \\ 3 & -5 & 10 & 2-\lambda & -1 \\ 26 & 85 & 50 & -1 & 138-\lambda \end{vmatrix} = -\lambda^5 + 284\lambda^4 - 14883\lambda^3 = 0$$

$$\lambda_1 = 69.3295$$

$$\lambda_2 = 214.6705$$

$$\lambda_3 = \lambda_4 = \lambda_5 = 0$$

4.3.3 (c) eigenvectors

Eigenvectors for $\mathbf{M}^T\mathbf{M}$:
When $\lambda_1 = -\sqrt{5281} + 142 = 69.3295$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \frac{68}{219} - \frac{\sqrt{5281}}{219} \\ -\frac{5}{73} - \frac{\sqrt{5281}}{73} \\ 1 \end{bmatrix} \approx \begin{bmatrix} -0.0213 \\ -1.0639 \\ 1 \end{bmatrix}$$

When $\lambda_2 = \sqrt{5281} + 142 = 214.6705$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \frac{68}{219} + \frac{\sqrt{5281}}{219} \\ -\frac{5}{73} + \frac{\sqrt{5281}}{73} \\ 1 \end{bmatrix} \approx \begin{bmatrix} 0.6423 \\ 0.9269 \\ 1 \end{bmatrix}$$

When $\lambda_3 = 0$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} -3 \\ 1 \\ 1 \end{bmatrix}$$

To verify with the results, here are the results calculated with `numpy.linalg.eig`:
When $\lambda_1 = -\sqrt{5281} + 142 = 69.3295$, $\lambda_2 = \sqrt{5281} + 142 = 214.6705$, $\lambda_3 = 0$

$$v_1 = \begin{bmatrix} -0.0146 \\ -0.7286 \\ 0.6848 \end{bmatrix}, v_2 = \begin{bmatrix} 0.4262 \\ 0.6150 \\ 0.6634 \end{bmatrix}, v_3 = \begin{bmatrix} 0.9045 \\ -0.3015 \\ -0.3015 \end{bmatrix}$$

Eigenvectors for $\mathbf{M}\mathbf{M}^T$:
When $\lambda_1 = -\sqrt{5281} + 142 = 69.3295$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} -\frac{58}{71} - \frac{\sqrt{5281}}{71} \\ \frac{1135}{568} + \frac{11\sqrt{5281}}{568} \\ -\frac{825}{284} - \frac{13\sqrt{5281}}{284} \\ -\frac{361}{568} - \frac{5\sqrt{5281}}{568} \\ 1 \end{bmatrix} \approx \begin{bmatrix} -1.8404 \\ 3.4056 \\ -6.2314 \\ -1.2753 \\ 1 \end{bmatrix}$$

When $\lambda_2 = \sqrt{5281} + 142 = 214.6705$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} -\frac{58}{71} + \frac{\sqrt{5281}}{71} \\ \frac{1135}{568} - \frac{11\sqrt{5281}}{568} \\ -\frac{825}{284} + \frac{13\sqrt{5281}}{284} \\ -\frac{361}{568} + \frac{5\sqrt{5281}}{568} \\ 1 \end{bmatrix} \approx \begin{bmatrix} 0.2066 \\ 0.5909 \\ 0.4215 \\ 0.0041 \\ 1 \end{bmatrix}$$

When $\lambda = 0$,

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{bmatrix} = \begin{bmatrix} -\frac{20}{7} \\ \frac{2}{7} \\ 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} -\frac{3}{7} \\ \frac{1}{7} \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -\frac{11}{7} \\ -\frac{8}{7} \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

To verify with the results, here are the results calculated with `numpy.linalg.eig`:
When $\lambda_1 = -\sqrt{5281} + 142 = 69.3295$, $\lambda_2 = \sqrt{5281} + 142 = 214.6705$, $\lambda = 0$

$$v_1 = \begin{bmatrix} 0.2449 \\ -0.4533 \\ 0.8294 \\ 0.1697 \\ -0.1331 \end{bmatrix}, v_2 = \begin{bmatrix} -0.1649 \\ -0.4716 \\ -0.3365 \\ -0.0033 \\ -0.7982 \end{bmatrix}, v_3 = \begin{bmatrix} -0.9553 \\ -0.0348 \\ 0.2708 \\ 0.0441 \\ 0.1037 \end{bmatrix}, v_4 = \begin{bmatrix} 0.0579 \\ 0.7363 \\ 0.3181 \\ -0.1271 \\ -0.5806 \end{bmatrix} v_5 = \begin{bmatrix} -0.0242 \\ -0.1456 \\ 0.1342 \\ -0.9792 \\ 0.0385 \end{bmatrix}$$

4.3.4 (d) SVD

For SVD decomposition,

$$\boldsymbol{M} = \boldsymbol{U\Sigma V}^{\mathrm{T}}$$

U consists of $\mathbf{MM}^T$'s eigenvectors and **V** consists of $\mathbf{M}^T\mathbf{M}$'s eigenvectors. $\mathbf{\Sigma}$ is a diagonal matrix with every element as the square root of eigenvalues. Since we have $\lambda_3 = 0$, we ignore any eigenvector related to it. Thus, matrix **U** reduces to:

$$\mathbf{U} = \begin{bmatrix} 0.2066 & -1.8404 \\ 0.5909 & 3.4056 \\ 0.4215 & -6.2314 \\ 0.0041 & -1.2753 \\ 1 & 1 \end{bmatrix}$$

Matrix $\mathbf{\Sigma}$ reduces to:

$$\begin{bmatrix} \sqrt{214.6705} & 0 \\ 0 & \sqrt{69.3295} \end{bmatrix}$$

Finally, matrix $\mathbf{V}^T$ reduces to:

$$\begin{bmatrix} 0.6423 & 0.9269 & 1 \\ -0.0213 & -1.0639 & 1 \end{bmatrix}$$

4.3.5 (e) One-dimensional approximation to M

After setting the smaller singular value to 0, we get:

$$\begin{bmatrix} \sqrt{214.6705} & 0 \\ 0 & 0 \end{bmatrix}$$

Using this new $\mathbf{\Sigma}$ to approximate $\mathbf{M} = \mathbf{U\Sigma V}^T$:

$$\begin{bmatrix} 1.9443 & 2.8057 & 3.0270 \\ 5.5608 & 8.0248 & 8.6576 \\ 3.9666 & 5.7242 & 6.1756 \\ 0.0386 & 0.0557 & 0.0601 \\ 9.4107 & 13.581 & 14.562 \end{bmatrix}$$