AMATH 563 Homework 03
Extended Yale Faces B Database – Eigenfaces
Joseph J. Williams
2020 May 27

In this study, the singular value decomposition (SVD) is briefly explored and applied to data consisting of cropped and uncropped images from the Yale Faces Database. Then, a variety of machine learning (ML) algorithms, including both supervised and unsupervised methods, are applied to the cropped images. The supervised methods include k-nearest neighbors, linear and quadratic discriminant analysis, Naive Bayes, and the support vector machine (SVM), while the only unsupervised method is k-means clustering. For the supervised methods, the results are cross-validated by partitioning the total data into training and testing sets and iterating k times (k-fold validation); the results are similarly cross-validated for the unsupervised algorithms. Finally, rather than work with the entire image space, the ML algorithms are applied only to a rank–r approximation of the feature space for computational ease; the value of r is informed by the SVD analysis. The supervised algorithms are used to identify individual faces and to classify images by gender, while the unsupervised algorithms are used to find patterns in the data. For both identifying faces and genders, both rank 400 and 1200 reconstructions, and both $k = 5$ and $k = 20$ repetitions, the diagonal linear discriminant algorithm is the best, with accuracies of >90%. For the unsupervised algorithms, the dependence of the KL divergence between the suggested and true label distribution is explored as a function of k and r for genders, subjects, and poses. The clusters suggested by k-means identify patterns in the lighting styles of the images.

Sec. I Introduction and Overview

There exist many scenarios in which one may possess a wealth of data but no rigorous model with which to understand it; such scenarios range from turbulent flows to data from social media networks or automatic image processing. *Machine learning* (ML) seeks to expose low-rank behavior in high-dimensional data, conveniently bypassing the need of a scientific model for prediction or exploitation. This study seeks to understand the Yale Faces Database, which consists of cropped and uncropped images of human faces. There are 165 uncropped images, consisting of 11 images each for 15 subjects with different poses, lightings, or accessories, and there are 2432 cropped images, consisting of 64 close-up images each for 38 subjects with different lightings and image qualities. As an introduction to the images of the faces, reference Fig. 1.1 below, which shows the average face from the cropped (*left*) and uncropped (*right*) data sets; reference Fig. 1.2 on the following page for the average male face (*left*), and average female face (*right*) from the cropped data set.

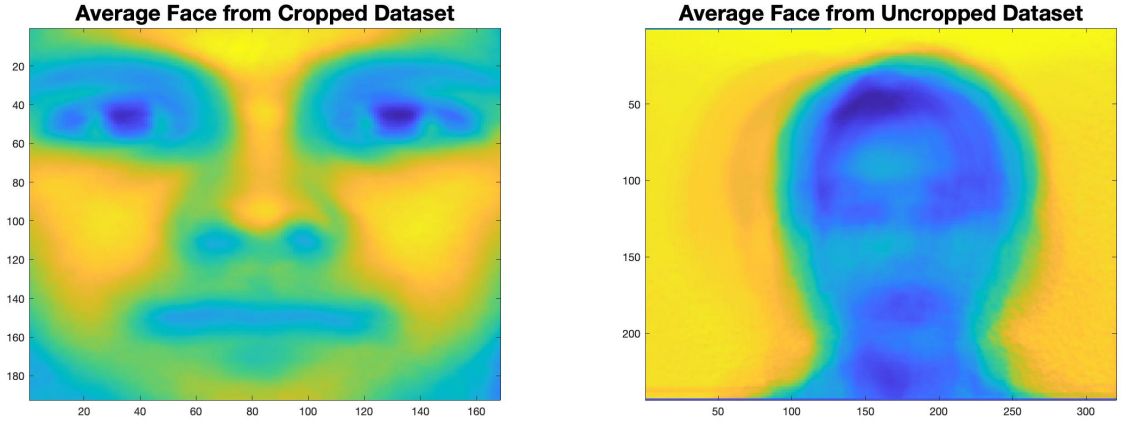

Figure 1.1: Average Face from Cropped (*left*) and Uncropped (*right*) Data Sets

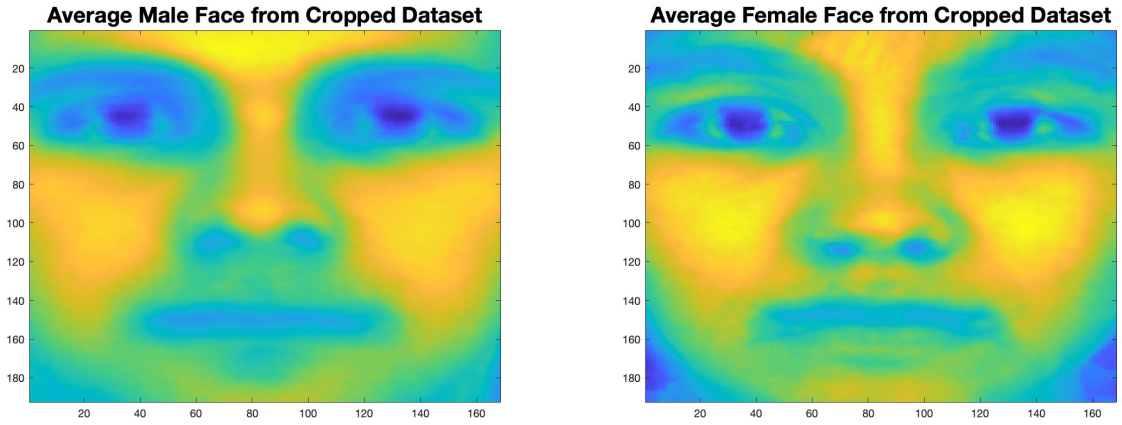

Figure 1.2: Average Male (*left*) and Female (*right)* Faces from Cropped

The data is first explored with the singular value decomposition (SVD), a matrix decomposition that is guaranteed to exist (unlike the eigendecomposition); the SVD of a matrix may be used to obtain a low-rank approximation to it. Even the most complex systems tend to generate data with low-rank structure, and the SVD of the data matrix may be used to find patterns from the data.

The data is then explored with various supervised and unsupervised ML algorithms. The supervised methods include k-nearest neighbors, linear and quadratic discriminant analysis, Naive Bayes, and the support vector machine (SVM); the unsupervised methods include k-means clustering. All algorithms will be defined and explained in § II. The supervised algorithms are used to identify individual faces and genders; supervised algorithms are first trained on a (usually large) subset of the data and then tested against the remaining, withheld data. The unsupervised algorithms are used to find patterns in the data that might not immediately be obvious with visual inspection.

In § II of this report, we will explore the theoretical background of the SVD and of ML algorithms, and in § III, we will discuss their algorithmic implementation. In § IV, we will discuss the results of applying the SVD to both the cropped and uncropped images before discussing which algorithms were better at identifying individual faces and genders; then we will explore the patterns in the images found by the unsupervised algorithms. § V will conclude this report with a summarization of what has been presented and discussed.

Sec. II Theoretical Background

The SVD is one of several decompositions of a matrix into multiple other matrices representing more fundamental characteristics of the matrix; other such decompositions are the QR decomposition, spectral (eigen) decomposition, and the LU decompositions, to name but a few. The SVD of $X \in \mathbb{R}^{m \times n}$ is:

$$X = U\Sigma V^*$$

Where $U \in \mathbb{R}^{m \times m}$, $\Sigma \in \mathbb{R}^{m \times n}$, $V \in \mathbb{R}^{n \times n}$ and V^* is the complex conjugate of V; there exists also the "economy" or "compact" version of the SVD, in which $U \in \mathbb{R}^{m \times r}$, $\Sigma \in \mathbb{R}^{r \times r}$, $V \in \mathbb{R}^{n \times r}$, where $r \leq \min\{m,n\}$ is the rank of M. U and V are unitary matrices with orthonormal columns, while Σ is a diagonal, non-negative matrix; V contains the feature space of the images. The columns of U are the left singular vectors and the columns of V are the right singular vectors of X. The entries of Σ are the singular values, or the singular value spectrum, of X; the number of nonzero singular values of X is equal to its rank. The singular values of X may also be understood to be the lengths of the semi-major axes of a hyperdimensional ellipse.

The SVD provides the optimal low-rank approximation of X. The optimal rank-r approximation to X, in a least-squares sense, is given by the rank-r SVD truncation of $\dot{X}$:

$$\text{armin}_{\dot{X},\ \text{s.t. rank}(\dot{X}) = r} \| X - \dot{X} \|_F = U_r \Sigma_r V_r^*,$$

where the subscript *r* denotes the first r leading columns of the matrix, and $\| \bullet \|_F$ is the Frobenius norm (Eckart-Young). With regards to our data matrix, which consists of pixel intensities of images, the importance of each feature to an individual image is given by the V matrix in the SVD; specifically, each column of V determines the loading, or weighting, of each feature onto a specific image (Brunton and Kutz). By plotting the singular value spectrum of X, we can see the importance of each mode to reconstructing X. The singular value spectrums of both the cropped and uncropped images will be presented in § IV.

In practice, there are a number of ways to determine how many modes to use for matrix reconstruction. One may inspect a plot of the singular value spectrum and choose to use the first *m* modes based on some visual criteria. By understanding the portion of energy contained within each mode as the singular values divided by the sum of the singular values, one may choose to use up to a certain percentage of the energy contained in the modes; finally, one may use modes with singular values above certain threshold τ; one such threshold is the optimal hard threshold as described on pg. 36 of Brunton and Kutz. Such methods will be demonstrated in § IV. The SVD analysis informs how many modes are required for good reconstruction for both the cropped and uncropped images. In the analysis using ML algorithms, rather than work with the full image space, we will work with a rank–*r* reconstruction of the image space for computational ease.

Next, we move to discussing ML algorithms. At a high level, ML algorithms apply regression techniques to the data for clustering and classification of different data types. There are two main types of ML algorithms: supervised and unsupervised. Supervised algorithms are trained on a set of data labeled or identified by an expert, finding by regression the best model that sorts between different data types; this model can then be used to identify new, unlabeled data. Variants of the basic supervised learning architecture include semi-supervised learning, active learning, and reinforcement learning. By contrast, unsupervised algorithms find patterns in the given data in a principled way; they are not given labels for the data but seek to find natural clusters in the data. Both supervised and unsupervised learning methods seek to create algorithms for classification, clustering, or regression.

The unsupervised and supervised algorithms used in this study will be discussed briefly in turn; for more details, including the mathematical formulation of these algorithms, the reader is referred to the textbook by Brunton and Kutz. Much of the following discussion is borrowed from that textbook.

Supervised:

1. Linear and Quadratic Discriminant Analysis (LDA and QDA): The goal of LDA is to find a linear combination of features that distinguishes the clusters; this amounts to projecting the data onto the line which maximizes the distance between the clusters. QDA allows this line to be a parabola, which can greatly increase the accuracy.
2. Support Vector Machines (SVM): The SVM algorithm seeks to draw decision lines through the data so that not only is the number of errors minimized, but also the largest margin between the data is optimized; this is, this method not only seeks accuracy but also eliminates edge cases.
3. Naive Bayes: The naive Bayes algorithm is a probabilistic classified based on Bayes' theorem. It assumes that the value of any particular feature of a cluster is independent from the values of all other features.

Unsupervised:

1. *k*–means: The k-means algorithm seeks to partition the given data into *k* clusters, where each data point is said to be in the cluster with the mean closest in value to its own; this involves computing the distances between two vectors repeatedly, and iteratively updating the values of the means. *k* is user-chosen.

In the pursuit of a model or predictive tool of any kind, cross-validation is critical. The supervised algorithms, by nature, require the data be split in training and testing sets, and that the training sets are already classified. In this study, we have only a single pool of the total data, already classified – there is no obvious way to split the data into training and testing sets; thus we randomly generate training and testing matrices from the matrix of the total data. The sizes of the training and testing sets may be varied, with larger training sets generally yielding better results. The accuracy of an ML algorithm is determined by comparing the actual labels of the testing data (which were known but withheld) with the labels suggested by the machine learning algorithms. The *accuracy* of the method is:

 # testing images correctly labeled / *# total testing images*.

However, because the training and testing sets in this study are randomly generated from a matrix of the total data (i.e. all images), the accuracy is specific to that particular training and testing set – a different training set would train the ML algorithm differently, and the testing set would be different, certainly resulting in different behavior and a different accuracy. Thus, we seek *k*-fold validation, in which the supervised ML algorithms are trained and tested with *k* different randomly generated training and testing subsets of the total data; the accuracy of the method is the average of the *k* different accuracies. This method of cross-validation will be used in this study.

For the unsupervised algorithms, cross-validation also takes the form of *k*-fold validation, in which the algorithm is repeated *k* times. The output of k-means is suggested label numbers of each data point; with this, a KL divergence between the suggested distribution of labels and the true distribution of labels is computed. Since our data is not truly unlabeled (i.e., we know the genders of each image, but we are choosing to use an unsupervised algorithm and feed it only the data). We will also investigate which images were grouped together in order to investigate how the algorithm clustered the images.

Separate from cross-validation strategies, one may compute various information criteria, such as the KL divergence and AIC and BIC scores, all discussed in the previous homework assignment. The KL divergence will be used to compare the distribution of suggested labels from the unsupervised k-means algorithm to the true distribution of labels for genders ($c = 2$), subjects ($c = 38$), and poses ($c = 64$). (In the images, the "poses" are different lighting positions and intensities.)

Sec. III Algorithm Implementation and Development

The rank and SVD of the cropped and uncropped matrices was computed with MATLAB's rank and *svd* commands: r_X = rank (X) and [U. Σ, V] = svd(X). From this, reconstructing the matrices with a certain rank using any criteria is straightforward. This also provides the feature space matrix V upon which many of the ML algorithms were performed, rather than the entire X matrix of the image data.

The implementation of the ML algorithms discussed in § II made use of the many in-built functions in MATLAB; reference Table 3 on the following page for the algorithms' in-built MATLAB functions.

Note that the diagonal quadratic discriminant method could not be used for identifying individual faces, as the algorithm used to randomly partition the data into training and testing sets could not always guarantee there would be at least three of each label in the testing data; similarly, the MATLAB in-built function for the naive Bayes algorithm could not be used for this task as it involved data with zero variance.

In order to run the algorithms, we had to create labels for the pictures, which for the first task were the subject numbers and for the second task were the genders. Creating the labels for the individuals was straightforward; for labeling the genders, the following subject numbers were identified as female, while the rest were male: 5, 15, 22, 27, 28, 32, 34, and 37. Note also that there were a few corrupted images; in addition in having to manually change the extensions from the totally useless .bad to .pgm, the true filetype of the images in this data set, these noisy samples were left in the dataset for the purposes of data augmentation.

Algorithm Name	In-Built MATLAB Function
Unsupervised	
k-means	kmeans(X_{total}, c)
Supervised	
k-nearest neighbors	
Diagonal Linear Discriminant	classify(X_{test}, X_{train}, C_{train}, 'diaglinear')
Diagonal Quadratic Discriminant	classify(X_{test}, X_{train}, C_{train}, 'diagquadratic')
Support Vector Machines	Face Identification: fitcecoc(X_{test}, X_{train}, C_{train}) Gender Classification: svm(X_{test}, X_{train}, C_{train})
Naive Bayes	fitcnb(X_{test}, X_{train}, C_{train})

Table 3: The in-built MATLAB functions used in this study.

For the supervised algorithms, the sizes of the training sets are varied from 5% to 95% of the total data. The training data is generated from the total data by randomly pulling data points and their labels; the unused data constitutes the testing data. *k*-fold validation was pursued by repeating this process *k* times, training and testing the algorithms on each of the *k* training and testing matrices, and finding the mean of the *k* accuracies. For the unsupervised algorithms, k-means was repeated *k* times, with a variable number of categories *c*. The accuracy of the supervised algorithms and KL divergence of the unsupervised algorithms were straightforward to compute. As mentioned previously, for computational speed and also to test the effect of different rank truncation values, all algorithms are run on a reduced version feature space vector V of size *n x r*, where *n* is the total number of images and *r* is the desired rank of the reconstructed data matrix.

Sec. IV Computational Results

We first discuss computational results regarding the SVD of our data before moving to the results of machine learning. The rank of the data matrix containing the cropped images was 2431, while the rank of the data matrix containing the uncropped images was 156. Reference Fig. 4.1 on the following page for the singular value spectrums and the cumulative energy plots of the cropped and uncropped images.

The amplitude of the first mode of the uncropped images is significantly larger than that of the cropped images. This is perhaps due to how close-up the cropped images are: because many of the faces are significantly different from each other, when the images include the background, there is significantly less variation from image to image, leading to a higher amplitude of the leading mode; conversely, when the images focus only on the faces, there is significantly more variation from image to image, leading to a lower amplitude of the leading mode. This is reflected in the cumulative energy plots: the cumulative energy rises more quickly for the uncropped images than it does for the cropped images, suggesting that larger modes hold less of the total energy for the uncropped images then for the cropped images. In turn, this suggests greater overall variation in the images.

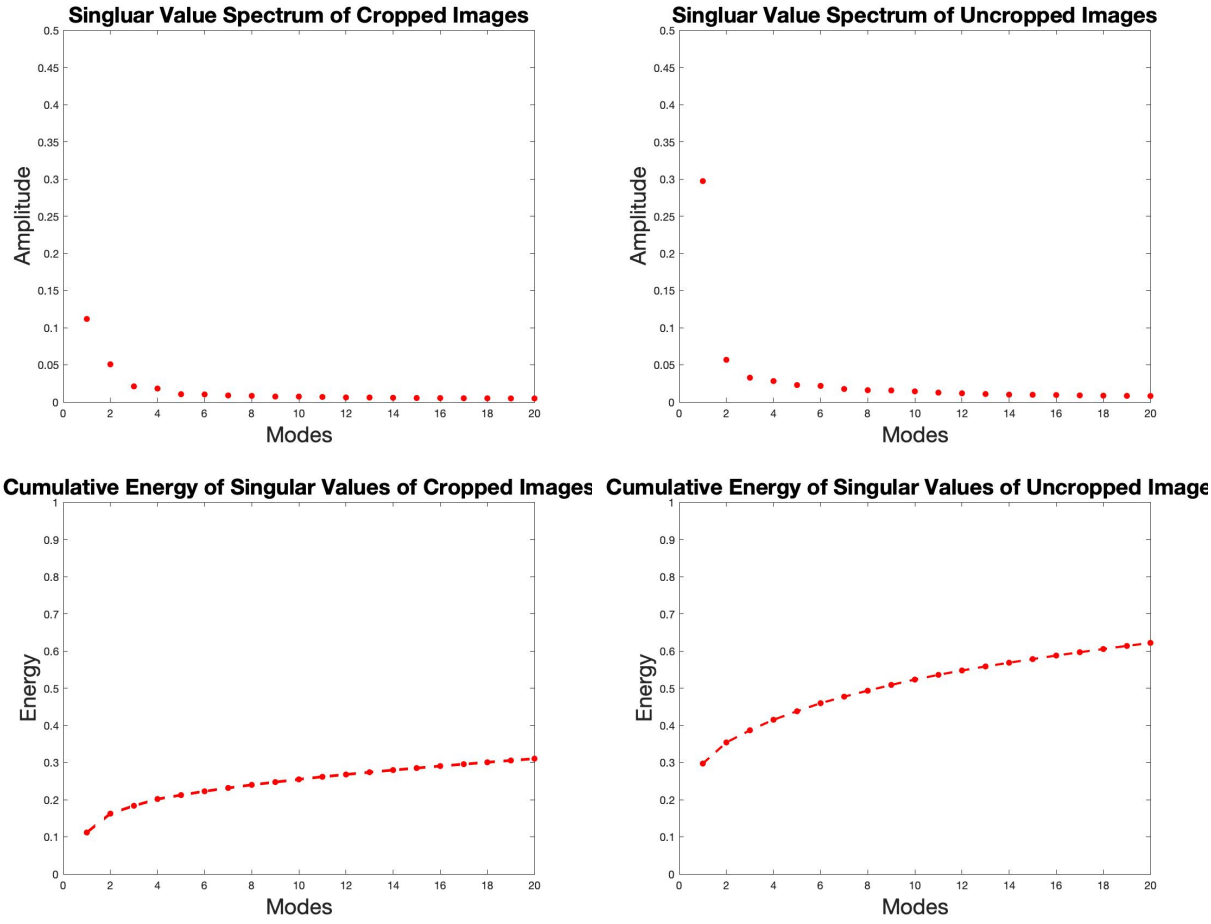

Figure 4.1: Singular Value Spectrums and Cumulative Energy Plots of Cropped and Uncropped Images

We next discuss image reconstruction. In the following discussion, for simplicity, we will be using only the first images from the cropped and uncropped image sets; using a given number of modes, the quality of reconstruction of this image is assumed to be the average quality of reconstruction of any image. Reference Fig. 4.2 on the following page, which shows the image reconstructions for the uncropped (*left*) and cropped (*right*) images using the first quarter, half, and three-quarter modes, and original (*bot*. to *top*).

Clearly, the cropped image is easier to reconstruct: with 25% of the modes, the uncropped reconstruction is dodgy but the subject in the cropped reconstruction is clearly identifiable. For the uncropped image, with 50% of the modes we are clearly able to recognize the original image, and with 75% of the modes, the reconstructed image is virtually indistinguishable from the original. As the number of modes increases for the cropped reconstruction, the quality continues to rise. All of the provided reconstructions look virtually the same.

Also note that reconstructions with the fewer modes capture only the most prominent features in the feature space, here: the mouth, nose, cheekbones, eyes, and hair. For the uncropped images, the face is a smaller proportion of the whole image, so we see that various outlines of heads are the main features, along with the lips, nose, eyes, and hair. For the cropped image, the main features generally do not include the head or facial hair or even eyebrows.

Before moving on to the results of ML algorithms, we consider reconstructing the images based on the percentage of cumulative energy up to a given mode. Reference Fig. 4.3 on pg. 8, which shows reconstructions using modes containing the first 25% (*top*) and 75% (*bot*.) of energy.

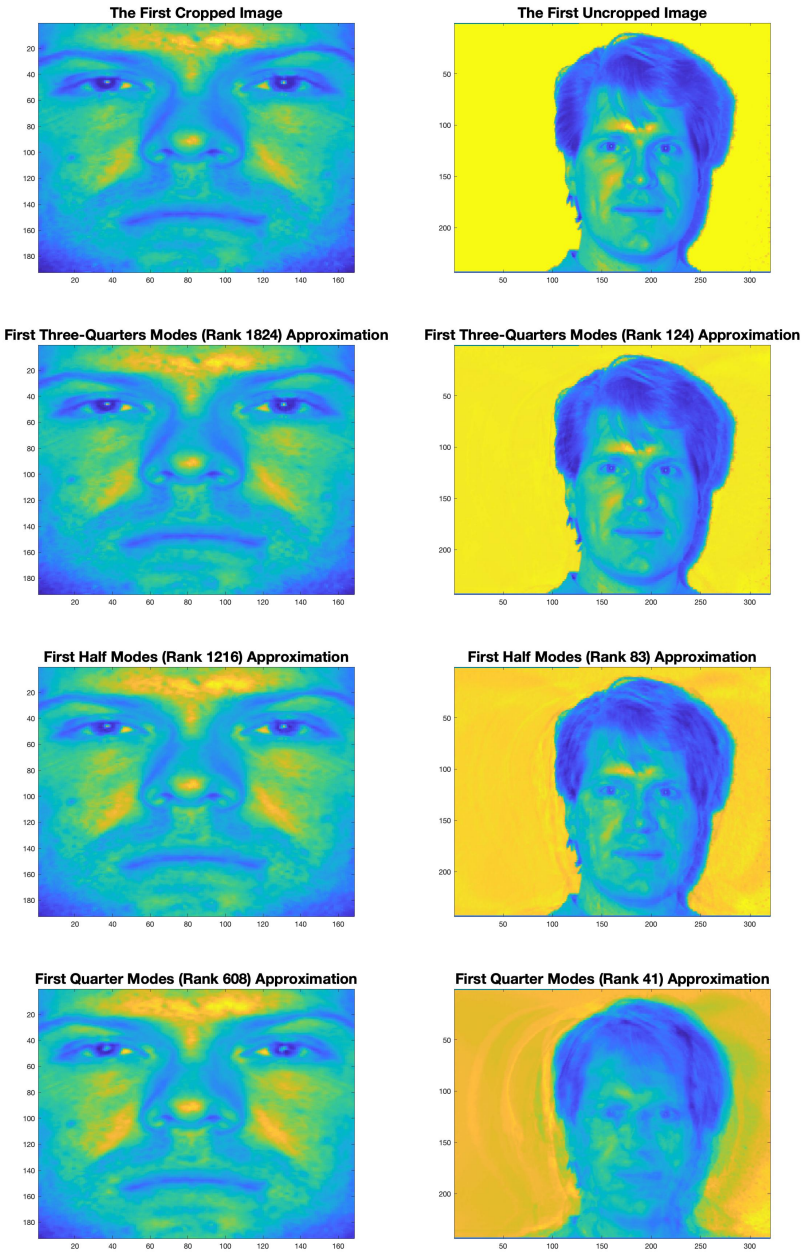

Figure 4.2: Uncropped (*left*) and Cropped (*right*) Image Reconstructions
for First Quarter, Half, and Three-Quarters Modes and Original (*bot. to top*)

As discussed previously, there is significantly more variation in the cropped images than in the uncropped images: the main features of a cropped image are the subject's eyes, nose, and mouth, whereas the main features of an uncropped images are the subjects face as a whole as well as the background. As such, the feature space tends to capture these defining features in a cropped image, so a low-energy reconstruction of a cropped image has more defining characteristics than a low-energy reconstruction of an uncropped image. On the other hand, the low-energy reconstruction of the uncropped image captures the most defining feature of those images: the outline of the head. It should be noted, however, that a 25% energy reconstruction of the uncropped image space is using only the first mode; considering what a downgrade this is from the full feature space, this is a fairly good reconstruction.

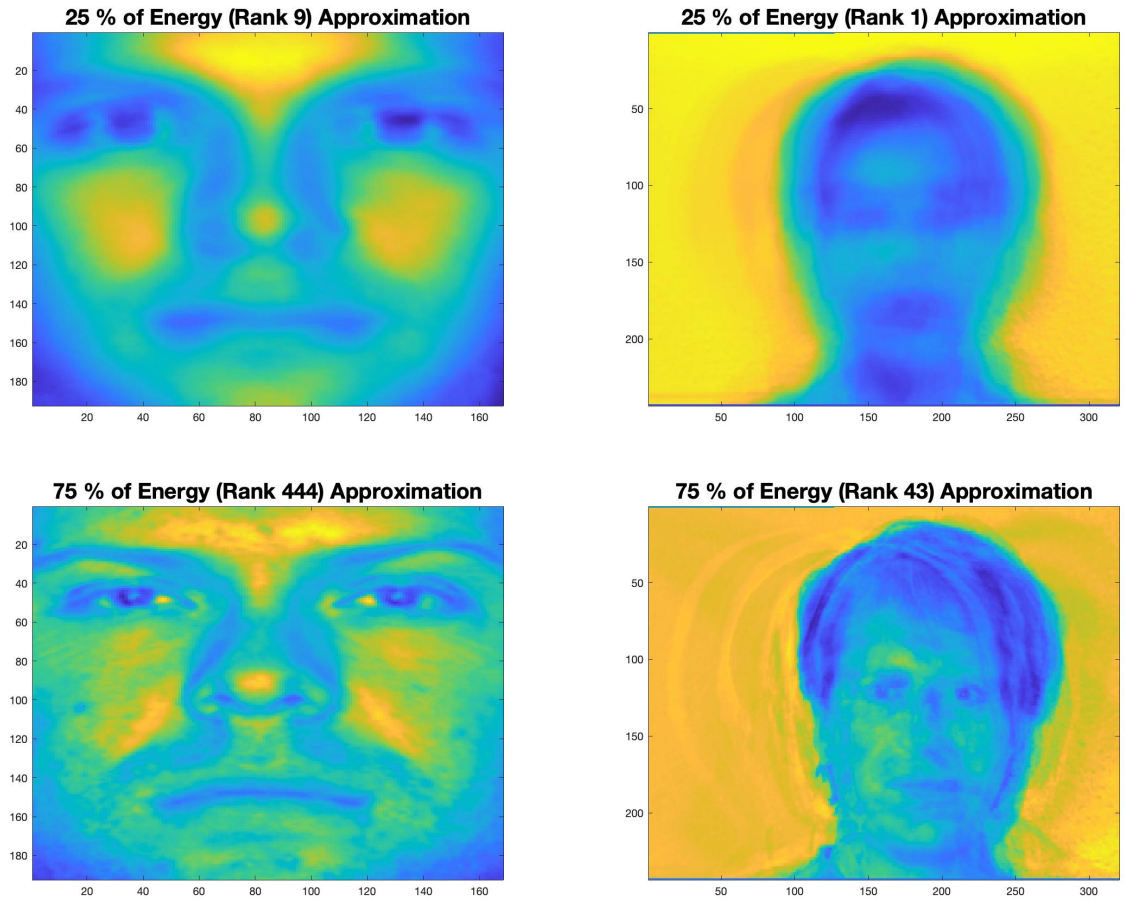

Figure 4.3: Cropped (*left*) and Uncropped (*right*) Image Reconstructions using 75% (*top*) and 95% (*bot.*) of the Total Energy

Based on the findings of Figs. 4.1 – 4.3, it appears that we need at least ~400 modes for good reconstruction of the cropped images and at least ~80 modes for good reconstruction of the uncropped images. Following this, for the analysis involving ML algorithms, we will work with rank 400 and rank 1200 truncations of the cropped image feature space. This closes our discussion of the SVD in this study.

We will now discuss the results of applying supervised and unsupervised ML algorithms to the cropped images. Reference Fig. 4.4 on the following page, which shows the accuracy vs. training size for the individual face identification task for rank $r = 400$ (*left*) and $r = 1200$ (*right*) reconstructions and using $k = 5$ (*top*) and $k = 20$ (*bot.*).

The diagonal linear discriminant clearly significantly outperforms SVM for all but the largest training set sizes. Additionally, the accuracy for the largest training set sizes is over 90% and close to 100% for the largest training set sizes, indicating an almost perfect identification algorithm. The accuracy's dependence on the training set size appears to be roughly the same for all combinations of k and r.

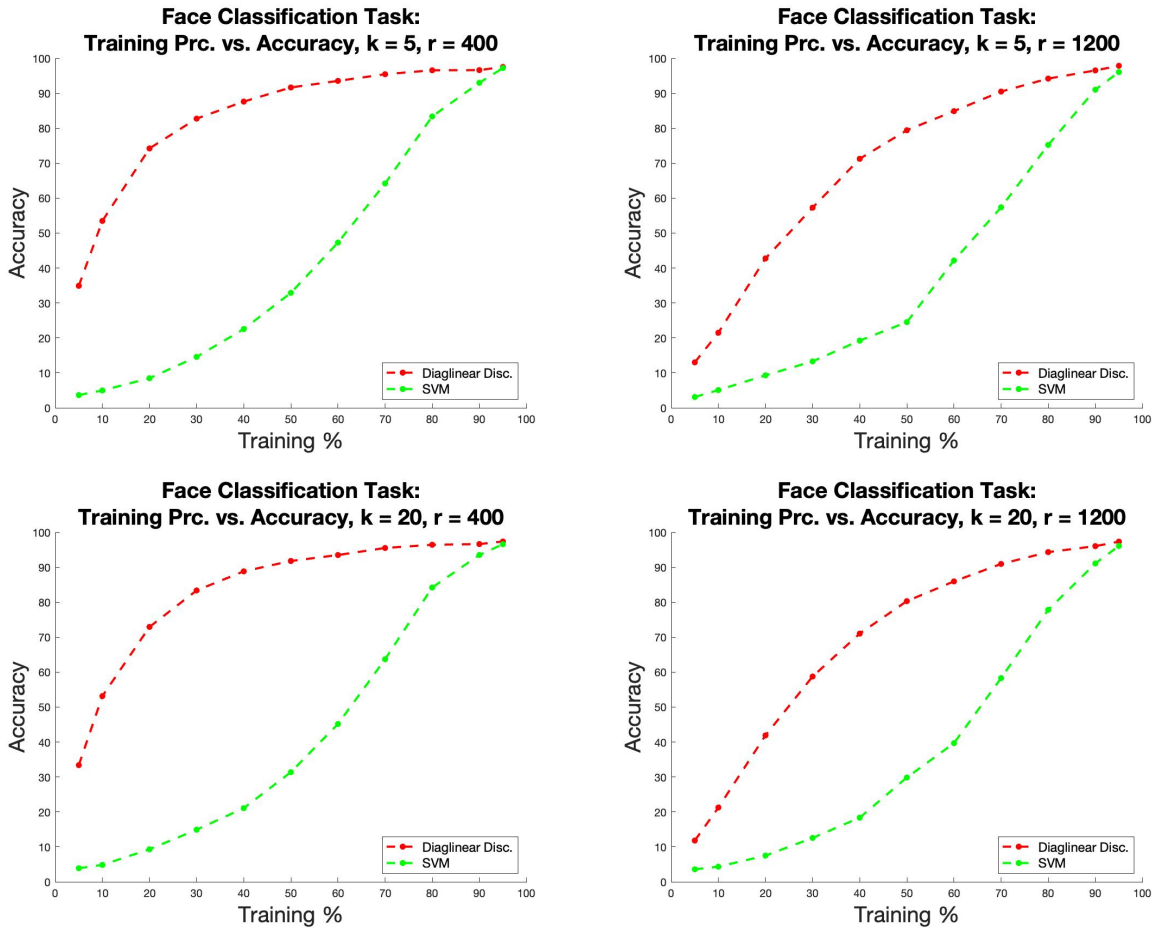

Figure 4.4: Accuracy vs. Size of Training Set, Individual Face Identification,
$r = 400$ (*top*) and $r = 1200$ (*bot.*) Reconstructions, $k = 5$ (*left*) and $k = 100$ (*right*) Repetitions

Moving to the gender classification task, reference Fig. 4.5 on the following page, which shows the accuracy vs. training size for rank $r = 400$ (*left*) and $r = 1200$ (*right*) reconstructions and using $k = 5$ (*top*) and $k = 20$ (*bot.*). When $k = 5$, the accuracy fluctuates seemingly randomly with training set size, which is to be expected; when we increase the repetitions to $k = 20$, we see more monotonic behavior in the accuracy, at least for the diagonal linear discriminant. It is very interesting that the diagonal quadratic discriminant and the naive Bayes algorithms perform almost exactly the same, even exhibiting the same unpredictable increases and decreases with training size; further, they both seem to actually do *worse* as training set size is increased for $r = 1200$ and do not seem to differ from $r = 400$.

In all cases, the diagonal linear discriminant performs best, with an accuracy of ~95% when the training set is 95% of the total (feature space) data; for $(r,k) = (400,20)$ has the highest accuracy at ~100%. That this case provides the highest accuracy is surprising to me, as it had the greatest k-fold validation and used the lower quality reconstructions of the images. Since the images were reconstructed with lower rank ($r = 400$, as opposed to $r = 1200$), they looked more uniform – I would think this would make the training less effective and make it more difficult to distinguish between images. On the other hand, since we are only distinguishing between male and female, and not individual faces, it is possible that by using a lower quality reconstruction, we are removing the details that distinguish individual faces while retaining details that distinguish between male and female faces.

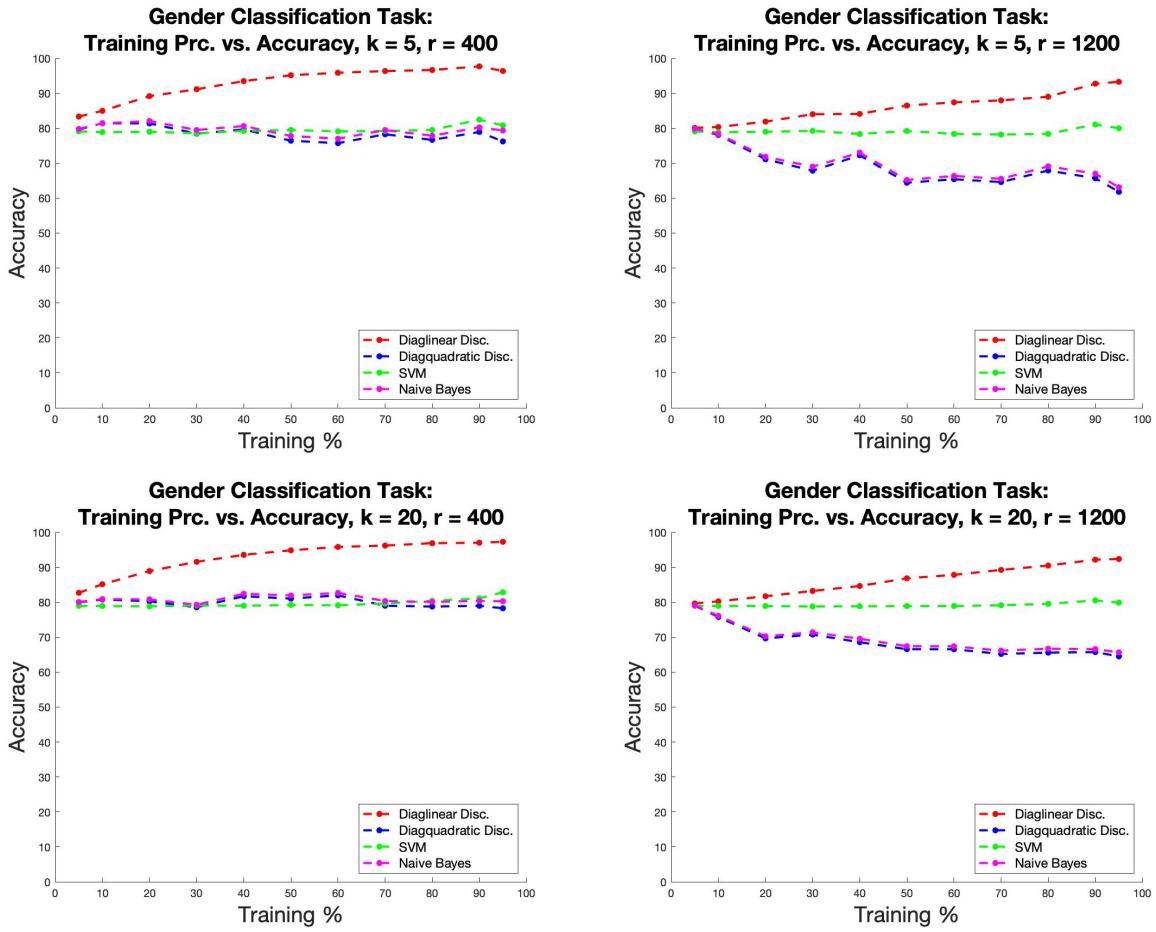

Figure 4.5: Accuracy vs. Size of Training Set, Gender Classification,
$r = 400$ (*top*) and $r = 1200$ (*bottom*) Reconstructions, $k = 5$ (*left*) and $k = 20$ (*right*) Repetitions

Finally, we move to discussing the results of the unsupervised algorithms. Recall that the k-means algorithm accepts unlabeled data and discovers natural clusters in the data. Our data is already labeled, however, giving us a ground truth with which to compare the proposed models generated from the unsupervised algorithm: the gender labels (1 to 2), individual face labels (1 to 38) and pose labels (1 to 64). Reference Fig. 4.6 below for bar charts of the distribution of the different labels in the data.

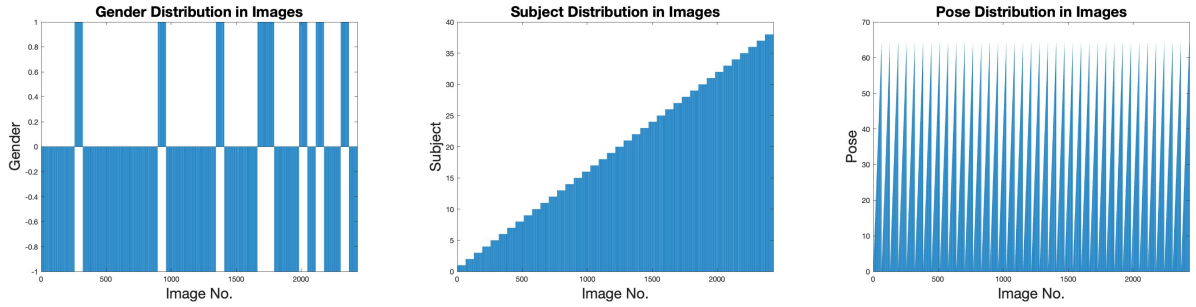

Figure 4.6: Gender, Subject, and Pose Distribution in the Cropped Images

Since we have a ground truth distribution available, we may compute the KL divergence between this ground truth and the distribution generated by the k-means algorithm; reference Fig. 4.7 on the following page, which shows the proposed distributions for gender, subject, and pose (*left to right*) for rank reconstructions $r = 5$ (*top*) $r = 100$ (*mid.*),

and $r = 2000$ (*bot.*) with their KL divergences, averaged across $k = 1000$ repetitions, and the standard deviation of the KL divergences.

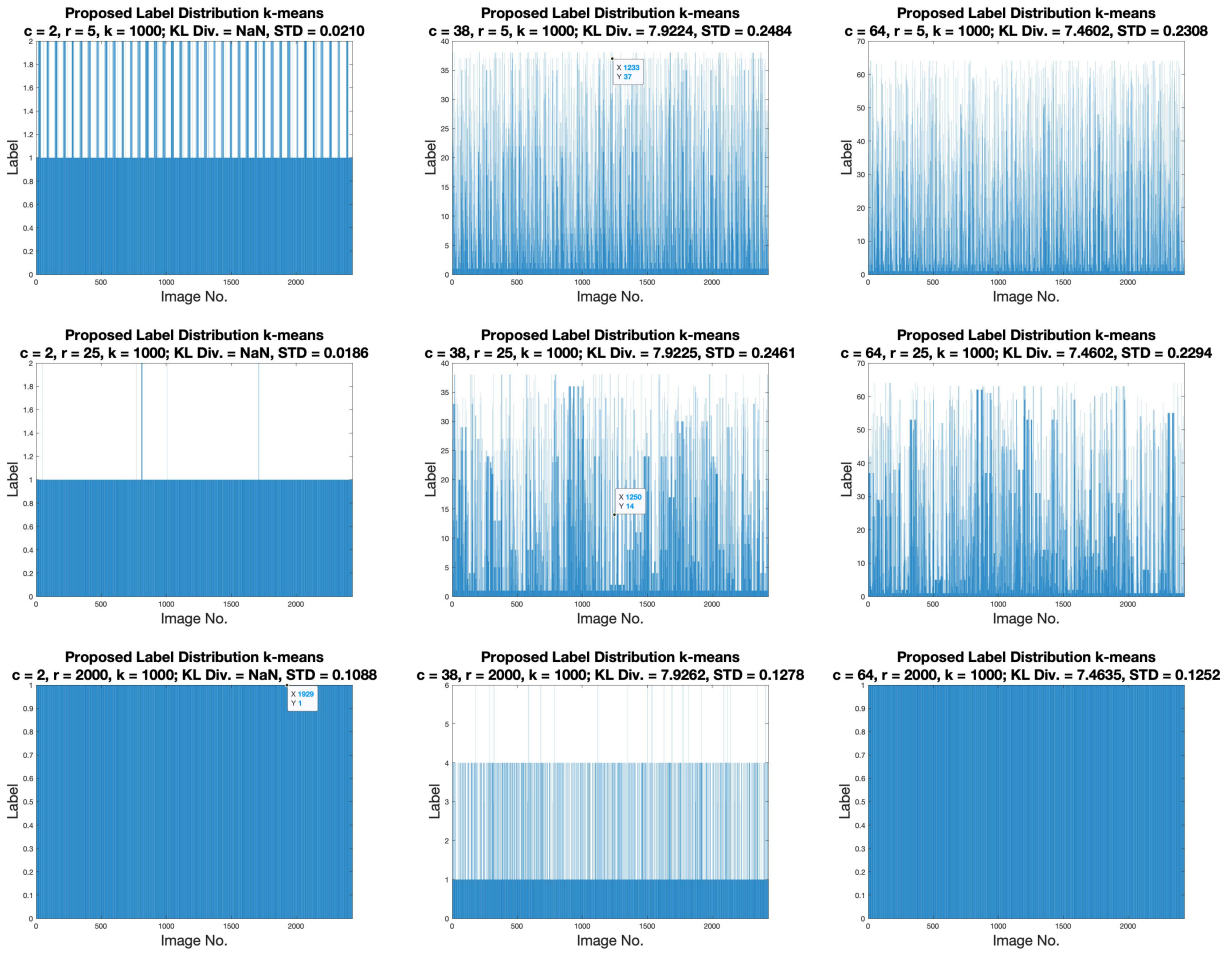

Figure 4.7: Proposed Distributions for Gender, Subject, and Pose (*left to right*) for Rank Reconstructions $r = 5$ (*top*) $r = 100$ (*mid.*), and $r = 2000$ (*bot.*)

The proposed distribution for the genders unfortunately never converged to anything within my computational limits regardless of the value of r – every time I ran the algorithm, it was different, with a different average KL divergence. The distributions for the subjects and poses do, however, converge to a stable distribution and KL divergence, with more interesting behavior for lower values of r (*top*). The value of the KL divergence is rather large, however, suggesting that k-means provides only a poor estimation of the true labels; this is in fact reflected in that, although the distributions of suggested labels do have a pattern, it differs significantly from the patterns in Fig. 4.6. The interesting pattern is picking up the different lighting angles and brightnesses of the photos, many of which have the same lighting style (i.e. shadow to one side of the face or other, or no shadow at all). This is in fact a subset of the larger "poses" category.

Sec. V Summary and Conclusions

This study briefly explored the significance and applications of the SVD of a data matrix before investigating the application of various ML algorithms, including both supervised and unsupervised, to the image data from the Yale Faces Database in order to classify the images based on individual faces and on gender, and to explore hidden features of the data.

When the SVD was applied to the data, it was found that at least ~400 modes were needed for good reconstruction of the cropped images and at least ~80 modes for good reconstruction of the uncropped images. Following this, for the analysis involving ML algorithms used rank 400 and rank 1200 trunctions of the cropped image feature space. This closes our discussion of the SVD in this study.

The SVD provides relevant and meaningful characterizations of the data, but the application of ML algorithms allows us to move beyond characterization and into application and prediction. Supervised methods were trained on a subset of the total data to classify images based on identified faces and on gender and then were tested on a withheld set of data. Unsupervised methods were similarly used to generate models for the data based on a certain number of clusters; the KL divergence of the models for different rank truncations were computed. It was found that the diagonal linear discriminant supervised algorithm had the greatest accuracy for both face and gender identification for all training subset sizes, outperforming diagonal quadratic discriminant, naive Bayes, and SVM. With regards to the unsupervised algorithms, k-means was not able to effectively cluster the images into any groupings that the supervised algorithms could, including gender, individual faces, or poses, but could effectively differentiate between the lighting styles and brightnesses.

With respect to the SVD analysis, future work includes computing the optimal hard threshold for image reconstruction, exploring reconstruction for images other than the first image in each set, exploring reconstruction techniques for other datasets of images (such as dogs and cats), and performing the SVD analysis on randomized subsets of the cropped and uncropped data matrices for optimization.

With respect to the ML algorithms, future work includes a more robust study of the dependence of accuracies and KL divergences on the rank truncation r, exploring the unsupervised dendrograms and the supervised k-nearest neighbors, decision tree algorithm, and random forest algorithm

Appendix A

This appendix provides a brief explanation of how the code runs.

The main code AMATH563_HW03_Main.m may be executed, and separate codes relating to the different sections of the homework assignment will be executed. It is straightforward.

Appendix B

See GitHub repository:

https://github.com/williamsj0165/AMATH563_HW03.git