# CIS 520, Machine Learning, Fall 2020
# Homework 5
# Due: Monday, November 2nd, 11:59pm
# Submit to Gradescope

Sheil Sarda

## 1 Perceptron vs. Winnow

### (a) Sparse target vector u, dense feature vectors $\mathbf{x}_t$.

Winnow Algorithm is a better choice. Because

$$||x_t||_2 = R_2 = \sqrt{d}$$

$$||u||_2 = \sqrt{k}$$

Also, we know $||u||_1 = k$ and $R_\infty = 1$. When we consider the upper bounds of the errors of Winnow Algorithm, we have

$$\frac{2ln(d)k^2}{\gamma^2}$$

Similarly, for Perceptron Algorithm:

$$\frac{dk}{\gamma^2}$$

Because $k << d$, we know $\frac{dk}{\gamma^2} >> \frac{2ln(d)k^2}{\gamma^2}$. Then Winnow Algorithm is a better choice.

### (b) Dense target vector u, sparse feature vectors $\mathbf{x}_t$.

Perceptron Algorithm is a better choice. Because

$$R_2 = \sqrt{k}$$

$$||u||_2 \leq 2\sqrt{d}$$

Also, we know $R_\infty = 1$ and $||u||_1 = d$. When we consider the upper bounds of the errors of Winnow Algorithm, we have

$$\frac{2ln(d)d^2}{\gamma^2}$$

Similarly, for Perceptron Algorithm:

$$\frac{4kd}{\gamma^2}$$

Because $k << d$, we know $\frac{2d^2ln(d)}{\gamma^2} >> \frac{4kd}{\gamma^2}$. Then Perceptron Algorithm is a better choice.

**(c) If your problem has non-negative feature vectors $\mathbf{x}_t \in \mathbb{R}^d_+$, is the Winnow algorithm a meaningful choice? Why or why not?**

No. Since the classification model $u$ of Winnow Algorithm is non-negative and $u^T x_t$ is always positive, when the data have the feature vectors $x_t \geq 0$. Then all the data are in the same class, indicating that Winnow is not a meaningful choice.

# 2 Singular Value Decomposition

1. Let $\mathbf{X}$ be a $n$ by $p$ matrix. Show that if $\mathbf{X}$ has a rank $p$ (all its columns are linearly independent), and $n > p$, then using the $p$-dimensional pseudo-inverse $\mathbf{X}^+ = \mathbf{V}_k \Lambda_k^{-1} \mathbf{U}_k^T$ in $\hat{\mathbf{w}} = \mathbf{X}^+ \mathbf{y}$ with $k = p$ solves the least squares problem $\hat{\mathbf{w}} = \arg\min_w (\mathbf{y} - X\mathbf{w})^T (\mathbf{y} - \mathbf{X}\mathbf{w})$.

   SVD: There is an orthogonal $n$ by $n$ matrix $U$, an orthogonal $p$ by $p$ matrix $V$, and an upper-diagonal $n$ by $p$ matrix $\Lambda$ whose top $p$ rows form a diagonal matrix with all the diagonal elements not zeros, because $X$ has rank $p$, and whose bottom $(n–p)$ rows are all zeros, such that $X = U\Lambda V^T$. The left pseudo-inverse of $\Lambda$ is a p by n matrix $\Lambda^{-1}$ which has its first p columns form a diagonal p by p matrix with its diagonal elements reciprocals of the respective elements of $\Lambda$ and the remaining $(n–p)$ columns made of zeros.
   Let us denote the p by p unit matrix $I_p$, the matrix formed by the top p rows of $\Lambda$ as $\Lambda_p$, and the one formed by the first p columns of $\Lambda^{-1}$ as $\Lambda_p^{-1}$. So

   $$\Lambda^{-1}\Lambda = \Lambda_p^{-1}\Lambda_p = I_p$$

   $$\Lambda^T \Lambda = \Lambda_p^2$$

   $$(\Lambda_p^2)^{-1}\Lambda^T = \Lambda^{-1}$$

   We want to minimize $f(w) = (y - Xw)^T (y - Xw)$. The derivatives vanish at the minimum: $0 = \frac{1}{2}\frac{\partial f}{\partial w} = X^T X \hat{\mathbf{w}}$, then $X^T y = X^T X \hat{\mathbf{w}}$. Thus,

   $$V\Lambda^T U^T y = V\Lambda^T U^T U \Lambda U^T \hat{\mathbf{w}} = V\Lambda^T \Lambda V^T \hat{\mathbf{w}} = V\Lambda_p^2 V^T \hat{\mathbf{w}}$$

   Finally, we have

   $$\hat{\mathbf{w}} = V(\Lambda_p^2)^{-1} V^T (V\Lambda_p^2 V^T \hat{\mathbf{w}}) = V(\Lambda_p^2)^{-1} V^T (V\Lambda^T U^T y) = V\Lambda^{-1} U^T y = X^+ y$$

2. Given the eigenvectors of $\mathbf{X}\mathbf{X}^T$ as $(\mathbf{u}_1, ..., \mathbf{u}_k)$ and corresponding eigenvalues as $(\lambda_i, ..., \lambda_k)$, give an expression for computing an eigenvector $\mathbf{v}_i$ of $\mathbf{X}^T\mathbf{X}$ in terms of $\mathbf{X}$, $\mathbf{u}_i$, and $\lambda_i$.
   We are given that $\lambda_i u_i = XX^T u_i$. Now we multiply $X^T$ on the LHS: $X^T(\lambda_i u_i) = \lambda_i (X^T u_i) = X^T (XX^T u_i) = X^T X (X^T u_i)$. We see that $X^T u_i$ is an eigenvector of $X^T X$ with eigenvalue $\lambda_i$ on condition that

   $$X^T u_i \neq 0$$

   This condition can only be violated when $\lambda_i = 0$ because $X^T u_i = 0$ and $XX^T u_i = \lambda_i u_i$.

3. Let $\mathbf{X}$ be a $n$ by $p$ matrix. Under what conditions (in terms of the relationship between $n$ and $p$) would the above calculation be an efficient way to find the largest eigenvectors of $\mathbf{X}^T\mathbf{X}$?

   If $n < p$, $XX^T$ is an n by n matrix and $X^T X$ is a p by p matrix, and it is easier to find eigenvalues of a matrix of smaller size.

# 3 Principal Component Analysis

## 3.1 Part 1: Comparing Principal Components

1. Report the eigenvectors and eigenvalues here.

   Eigenvectors of the dataset

   - Eigenvector 1: $[0.70711, 0.70711]$
   - Eigenvector 2: $[-0.70711, 0.70711]$

   Eigenvalues of the dataset

   - 1.6530
   - 0.3583

2. Express mathematically, explain why the first PC is the eigenvector associated with the largest eigenvalue?

3. What can you say about the relationship between the first principal component and the second?

   The two principal components are orthogonal, by construction.

## 3.2 Part 2: Plotting Principal Components in Original Space

1. Please describe how the principal components relate to the points.

   The first principal component is the direction of maximum variance whereas the second principal component is the vector orthogonal to the first in the 2D plane.

2. Paste the graph here of the plot of the given points (with both axis in same scale) as well as the lines representing the principal components in original space, with x1 in the x axis and x2 in the y axis.

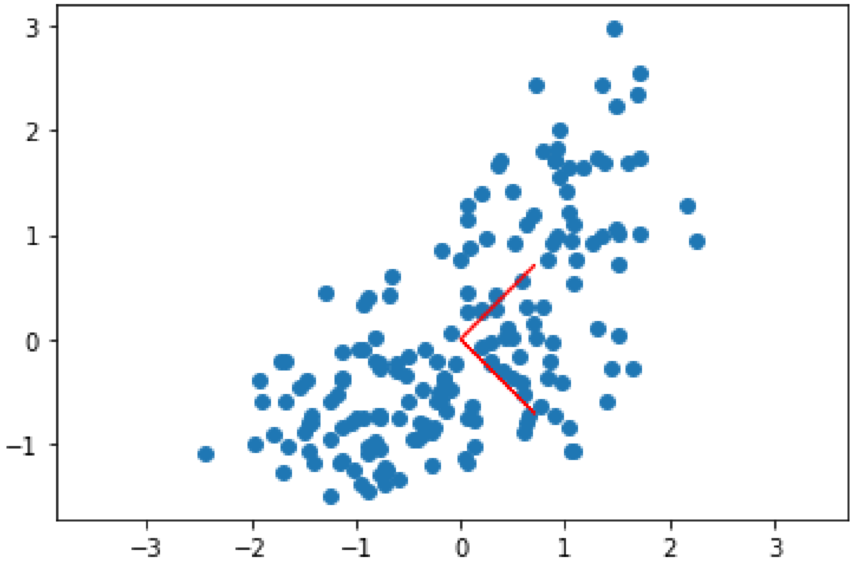

Figure 1: Scatter Plot with Principal components in the original space.

### 3.3 Part 3: Plotting Data Projected onto Component Space

1. Explain how the graph of points on principal component space relates to the graph of points on original space above.

   The graph of points in principal component space is simply the graph of points in the origin space, but rotated counter-clockwise by approximately 45°.

2. Explain the difference in distribution of points projected on the first component vs. projected on the second.

   The distribution of points projected on the first component has maximal variance compared to the distribution of points projected onto the second component.

3. Paste the plot of the given points (with both axis in same scale) in principal component space.

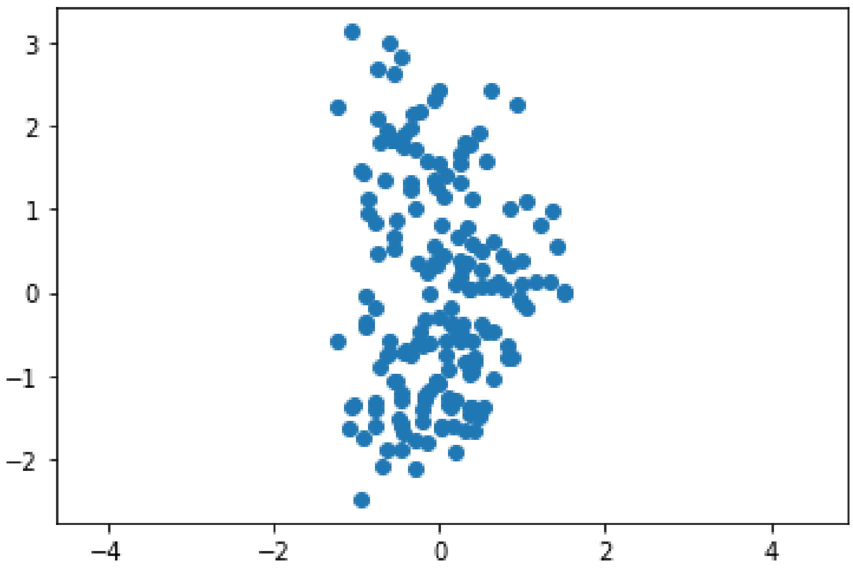

Figure 2: Scatter plot in Principal Component space.

### 3.4 Part 4: PCA and Reconstruction Error

1. Using the digit data set from the PCA worksheet (used in PCA Maximize Variance), what is the reconstruction error using the first and second principal components

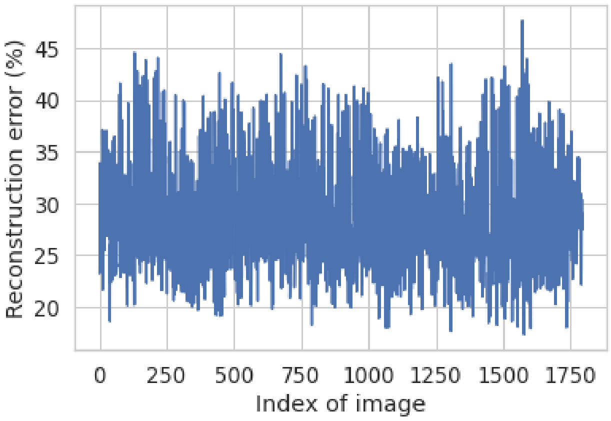

Figure 3: Reconstruction error of each input image using 2 principal components.

Average reconstruction error is 28.85%

2. Mathematically express how you come up with the answer, and explain how PCA is minimizing the reconstruction error.

Using Lagrangian multipliers, there exists $\lambda_{k+1}, \ldots, \lambda_d$ such that solution to above is given by:

$$\min \sum_{t=1}^{n} \sum_{j=k+1}^{d} \mathbf{w}_j^\top \Sigma \mathbf{w}_j + \sum_{j=k+1}^{d} \lambda_j \|\mathbf{w}_j\|_2^2 \qquad d \text{ is dim of the covar matrix, } K \text{ is number of features.}$$

Setting derivate to 0, $\Sigma \mathbf{w}_j = \lambda_j \mathbf{w}_j$. That is $\mathbf{w}_j'$s are eigenvectors and $\lambda_j$ 's are eigenvalues.

# 4 Principal Component Analysis on Faces

## 4.1 Part 1: PCA with SVD and Resulting Eigenfaces

1. To check the outputs of your PCA functions, report the singular values here.

| **Singular Values for PCA on Face dataset** | | | | | |
|---|---|---|---|---|---|
| 86.701965 | 66.46528 | 50.15517 | 39.722527 | 33.757385 | 31.568754 |
| 27.678606 | 25.354534 | 24.862415 | 22.975147 | 22.44062 | 21.298525 |
| 19.838667 | 19.02967 | 18.317488 | 17.568378 | 17.033203 | 16.045595 |
| 15.42673 | 15.356074 | 14.85018 | 13.929341 | 13.577002 | 13.410842 |
| 13.130961 | 12.957497 | 12.735859 | 12.511106 | 12.019814 | 11.801411 |
| 11.265183 | 11.012788 | 10.689299 | 10.276611 | 10.056747 | 9.988401 |
| 9.814747 | 9.709447 | 9.4375105 | 9.3004675 | 9.055668 | 8.954327 |
| 8.786291 | 8.701313 | 8.537433 | 8.454362 | 8.376851 | 8.342043 |
| 8.121107 | 8.044209 | | | | |

2. Please describe what the eigenfaces look like. What do you expect to observe with the eigenfaces associated with lower eigenvalues?

The eigenfaces represent abstractions of human features which explain the most variance within the dataset of images. The variance explained by each eigenface successively decreases for each eigenface, by definition of PCA.

3. Please insert your eigenfaces output here.

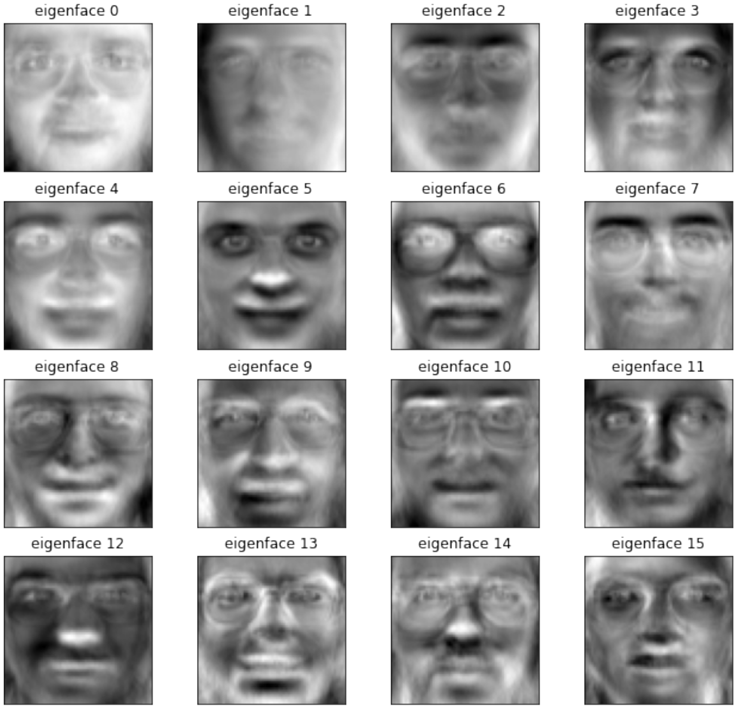

Figure 4: Eigenfaces.

## 4.2 Part 2: Reconstructing Faces

1. Paste in the reconstructed faces plot. Compare the reconstructed images to the original images. How are they similar and how are they different? Shortly explain why they are different?

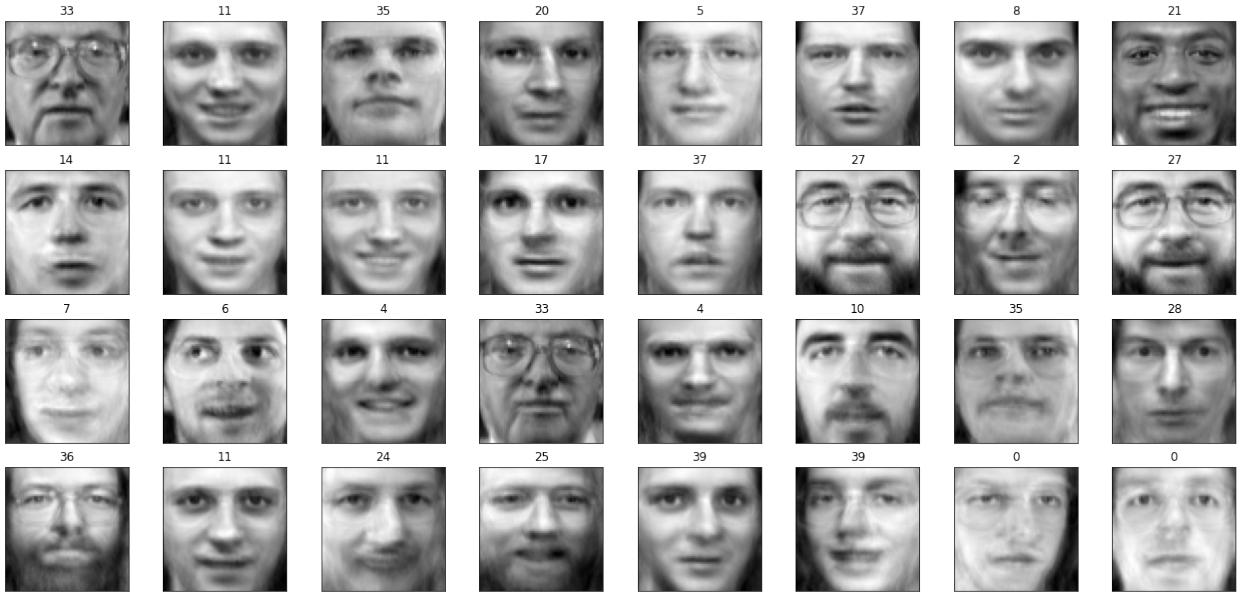

Figure 5: Reconstructed faces.

Similarities: The reconstructed images retain defining characteristics and structure of the human face. Differences:

(a) The reconstructed faces have lost variance in some features such as width and roundness of face, definition of nose, etc.

(b) Lost nuances in facial expressions and pose such as jaw position.

(c) New features such as definition of glasses got added on to faces even when the original image did not feature glasses.

2. What do you expect to see from the reconstructed images as the number of principal components chosen for PCA increases / decreases? Please explain why.

   As the number of principal components increases, the reconstructed faces start to resemble the original faces to a greater degree, since the Principal component space starts to resemble the original data space to a closer degree.

   Since PCA is a lower dimensional representation of the original data space, some clarity of the images will still be lost, but the number of principal components controls how close the reconstructed images will be to the original images.

## 4.3 Part 3: Variance Explanation

1. How do you expect (based on theory; please be precise!) the plot of variance explained as the number of components to relate to the eigenvalues of the corresponding components?

   The plot of cumulative variance explained is directly related to the eigenvalues of the corresponding components added to the model.

   This is mathematically explained by the fact that the eigenvalue of a principal component is a measure of the variance described by that component.

2. What is the relation between reconstruction error and the variance explained?

Reconstruction error is inversely correlated to explained variance since as the explained variance increases, the principal component space captures more variance of the original dataset, reducing the reconstruction error.

3. Insert the three line plots of explanation vs. number of components, descending eigenvalues vs. number of components, and reconstruction error vs. number of components here.

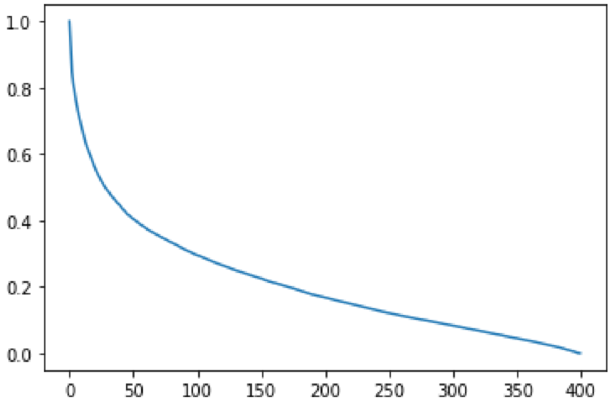

Figure 6: Explanation vs. number of components.

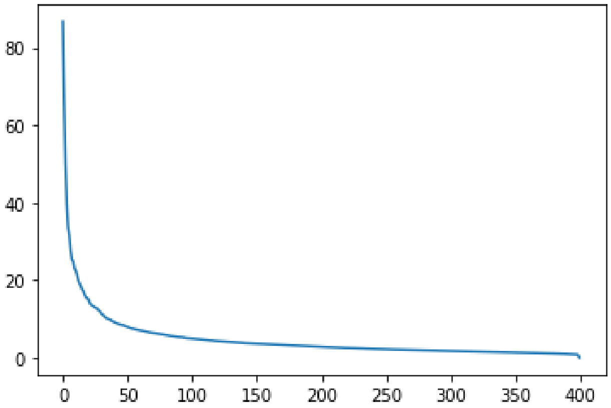

Figure 7: Descending eigenvalues vs. number of components.

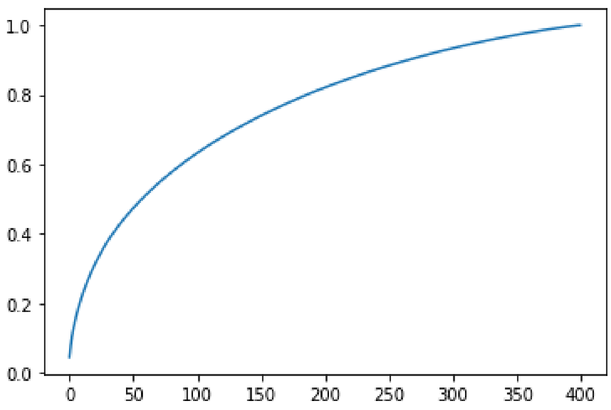

Figure 8: Reconstruction error vs. number of components.