CS 461 Homework Two

Matthew McCaughan

Problem 1

1.1 Data Matrix (4x4)

The linear model is formed using the given data points:

$$\Phi = \begin{pmatrix} 1 & 4 & 1 & 1 \\ 1 & 7 & 0 & 2 \\ 1 & 10 & 1 & 3 \\ 1 & 13 & 0 & 4 \end{pmatrix}$$

1.2 Exact or MMSE Solution?

The normal equation for linear regression is given by:

$$\Phi^T \Phi \mathbf{w} = \Phi^T \mathbf{y}$$

This equation provides the exact solution if $\Phi^T \Phi$ is invertible. If $\Phi^T \Phi$ is not invertible,it implies that the data is linearly dependent.

1.3 Is $\Phi^T \Phi$ Invertible? Solve for w

$$\Phi^T = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 4 & 7 & 10 & 13 \\ 1 & 0 & 1 & 0 \\ 1 & 2 & 3 & 4 \end{pmatrix}$$

$$\Phi^T \Phi = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 4 & 7 & 10 & 13 \\ 1 & 0 & 1 & 0 \\ 1 & 2 & 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 4 & 1 & 1 \\ 1 & 7 & 0 & 2 \\ 1 & 10 & 1 & 3 \\ 1 & 13 & 0 & 4 \end{pmatrix}$$

This will give a 4x4 matrix. After calculating $\Phi^T \Phi$, the determinant is calculated:

$$\det\left(\Phi^T \Phi\right) \approx 0$$

Since the determinant is zero, $\Phi^T \Phi$ is not invertible. Solving for w is done with the a pseudo inverse
Using numpy to solve for this pseudo inverse gives:

$$\mathbf{w} = \begin{pmatrix} 0 \\ 3 \\ 3 \\ 1 \end{pmatrix}$$

This gives the model:

$$y = 0 \cdot 1 + 3 \cdot x_1 + 3 \cdot x_2 + 1 \cdot x_3$$

1.4 Original Model

The original given model is:

$$y = 1 + 2 \cdot x_1 + 3 \cdot x_2 + 4 \cdot x_3$$

The estimated model from the data is:

$$y = 3 \cdot x_1 + 3 \cdot x_2 + 1 \cdot x_3$$

These models are not the same. This may be due to a lack of provided data which could lead to variation between models, as both fit the current data, but the provided model might be applicable for unforeseen data, where the estimated model applies only to the data that has been given.

1.5 New Data

$$\Phi = \begin{pmatrix} 1 & 4 & 1 & 1 \\ 1 & 7 & 0 & 2 \\ 1 & 10 & 1 & 3 \\ 1 & 13 & 0 & 4 \\ 1 & 16 & 1 & 5 \\ 1 & 19 & 0 & 6 \\ 1 & 22 & 1 & 7 \\ 1 & 25 & 0 & 8 \end{pmatrix}$$

Recalculating inverse to get new weights:

$$w = \begin{pmatrix} -0.039 \\ 2.995 \\ 3.083 \\ 1.011 \end{pmatrix}$$

The new model is now: $y = -0.039 \cdot 1 + 2.995 \cdot x_1 + 3.083 \cdot x_2 + 1.011 \cdot x_3$
This model is different from the previous one, the weights are similar, but differ slightly. Obtaining the original model from the regression method becomes more feasible as you accumulate more data to recover it from.

1.6 Column Deletion

It seems the third column only tends to increase with the number of data points collected, and seems redundant with column 1 as it also tends to increase as the number of data points increases, so this column would be a good candidate for removal

Problem 2

2.1 using MMSE

The problem provides two data points as example:

$$d_1 : (x_1, x_2) = (1, 0),\ y = 1$$

$$d_2 : (x_1, x_2) = (0, 1),\ y = 1$$

The task is to estimate the model $y = w_0x_1 + w_1x_2$ using the min mean squares error function The Min Mean Squared Error function is given by:

$$J(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{2}(y_i - (w_0x_{i1} + w_1x_{i2}))^2$$

Substituting the given data points obtains our equation:

$$J(\mathbf{w}) = \frac{1}{2}\left[(1 - w_0)^2 + (1 - w_1)^2\right]$$

In order to minimize the function, it is required to take partial derivates with respect to each (w$_i$),where each w$_i$ is a weight from the model. Below is each partial derivative taken:

$$\frac{\partial J(\mathbf{w})}{\partial w_0} = -(1 - w_0) = 0 \quad \Rightarrow \quad w_0^* = 1$$
$$\frac{\partial J(\mathbf{w})}{\partial w_1} = -(1 - w_1) = 0 \quad \Rightarrow \quad w_1^* = 1$$

Therefore the optimal solution is:

$$w_0^* = 1 \quad w_1^* = 1$$

2.2 Lagrangian Function

The constrained optimization problem introduces a constraint $||\mathbf{w}||^2 \leq C$. The Lagrangian function would be defined as:

$$\mathcal{L}(\mathbf{w}, \lambda) = J(\mathbf{w}) + \lambda\left(w_0^2 + w_1^2 - C\right)$$

And then after substituting the MMSE function $J(\mathbf{w})$, the Lagrangian becomes:

$$\mathcal{L}(\mathbf{w}, \lambda) = \frac{1}{2}\left[(1 - w_0)^2 + (1 - w_1)^2\right] + \lambda(w_0^2 + w_1^2 - C)$$

2.3 KKT Conditions and Optimal Solution

The KKT conditions provide the following:
1. Primal feasibility: $w_0^2 + w_1^2 \leq C$ 2. Stationarity: $\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, \lambda) = 0$ 3. Complementary slackness: $\lambda(w_0^2 + w_1^2 - C) = 0$ 4. Dual feasibility: $\lambda \geq 0$
To find the stationary points, take the partial derivatives of $\mathcal{L}$ with respect to w_0 and w_1:

$$\frac{\partial \mathcal{L}}{\partial w_0} = -(1 - w_0) + 2\lambda w_0 = 0$$
$$\frac{\partial \mathcal{L}}{\partial w_1} = -(1 - w_1) + 2\lambda w_1 = 0$$

Solving these gives:

$$w_0 = \frac{1}{1 + 2\lambda}, \quad w_1 = \frac{1}{1 + 2\lambda}$$

Applying the constraint $w_0^2 + w_1^2 = C$:

$$2\left(\frac{1}{1+2\lambda}\right)^2 = C$$

The optimal values of w_0^* and w_1^* are:

$$w_0^* = w_1^* = \frac{1}{1+2\lambda}$$

2.4 Results for Different Values of C

For different values of $C = \{0.5, 1, 2, 3\}$, the corresponding λ and w_0^*, w_1^* can be computed. Given the relationship between λ and C:

$$\lambda = \frac{1}{2}\left(\frac{\sqrt{2}}{\sqrt{C}} - 1\right)$$

and the optimal weights:

$$w_0^* = w_1^* = \frac{1}{1+2\lambda}$$

compute the values of w_0^* and w_1^* for different values of C.

1. $C = 0.5$

$$\lambda = \frac{1}{2}\left(\frac{\sqrt{2}}{\sqrt{0.5}} - 1\right) = \frac{1}{2}\left(\frac{\sqrt{2}}{0.7071} - 1\right) = \frac{1}{2}(2-1) = 0.5$$

$$w_0^* = w_1^* = \frac{1}{1+2\times 0.5} = \frac{1}{2} = 0.5$$

2. $C = 1$

$$\lambda = \frac{1}{2}\left(\frac{\sqrt{2}}{\sqrt{1}} - 1\right) = \frac{1}{2}(1.4142-1) = \frac{1}{2}(0.4142) = 0.2071$$

$$w_0^* = w_1^* = \frac{1}{1+2\times 0.2071} = \frac{1}{1.4142} \approx 0.7071$$

3. $C = 2$

$$\lambda = \frac{1}{2}\left(\frac{\sqrt{2}}{\sqrt{2}} - 1\right) = \frac{1}{2}(1-1) = 0$$

$$w_0^* = w_1^* = \frac{1}{1+2\times 0} = 1$$

4. $C = 3$

$$\lambda = \frac{1}{2}\left(\frac{\sqrt{2}}{\sqrt{3}} - 1\right) = \frac{1}{2}\left(\frac{1.4142}{1.7321} - 1\right) = \frac{1}{2}(0.8165-1) = \frac{1}{2}\times(-0.1835) = -0.0917$$

$$w_0^* = w_1^* = \frac{1}{1+2\times(-0.0917)} = \frac{1}{1-0.1835} = \frac{1}{0.8165} \approx 1.2247$$

C = 2 seems to be the optimal C because it follows the given data and would have no tendency to be overly complex or too simple

Problem 3

Recovery of Sinusoidal Function

3.1 Average validation error

Average Validation Error (OLS): 0.6877763487693266

3.2 selection of best λ

OLS Weights:

$$\mathbf{w}_{OLS} = \begin{bmatrix} 1.57599147 \\ -5.08761652 \times 10^{1} \\ 7.75713301 \times 10^{2} \\ -5.58640339 \times 10^{3} \\ 2.31257296 \times 10^{4} \\ -5.87488231 \times 10^{4} \\ 9.22298569 \times 10^{4} \\ -8.68934354 \times 10^{4} \\ 4.49598093 \times 10^{4} \\ -9.81384238 \times 10^{3} \end{bmatrix}$$

Best Lambda for Ridge:

$$\lambda^* = 0.00025595479226995333$$

Ridge Weights (Best λ):

$$\mathbf{w}_{ridge} = \begin{bmatrix} 0.08913726 \\ 6.43695864 \\ -12.64628367 \\ -6.00193829 \\ 3.56053704 \\ 7.37800967 \\ 6.18791708 \\ 2.47393524 \\ -1.8845177 \\ -5.85529829 \end{bmatrix}$$

3.4 Two test MSEs

Test MSE for OLS Model:

$$\text{MSE}_{OLS} = 0.05817688194803842$$

Test MSE for Ridge Model:

$$\text{MSE}_{ridge} = 0.03328579470315911$$

3.5 Large data set

OLS Weights (Large Dataset):

$$\mathbf{w}_{OLS,large} = \begin{bmatrix} -7.57590832 \times 10^{-2} \\ 1.35547773 \times 10^{1} \\ -1.43352321 \times 10^{2} \\ 1.24150706 \times 10^{3} \\ -6.00402894 \times 10^{3} \\ 1.60246664 \times 10^{4} \\ -2.48793568 \times 10^{4} \\ 2.24690829 \times 10^{4} \\ -1.09346097 \times 10^{4} \\ 2.21250061 \times 10^{3} \end{bmatrix}$$

OLS Predictions (Large Dataset):

$$\mathbf{y}_{OLS,large} = \begin{bmatrix} -0.07575908 \\ 0.67747791 \\ 1.06694914 \\ 0.90918982 \\ 0.31373938 \\ -0.339005 \\ -0.86655682 \\ -1.07249867 \\ -0.67936018 \\ -0.111753 \end{bmatrix}$$

Controlling Complexity

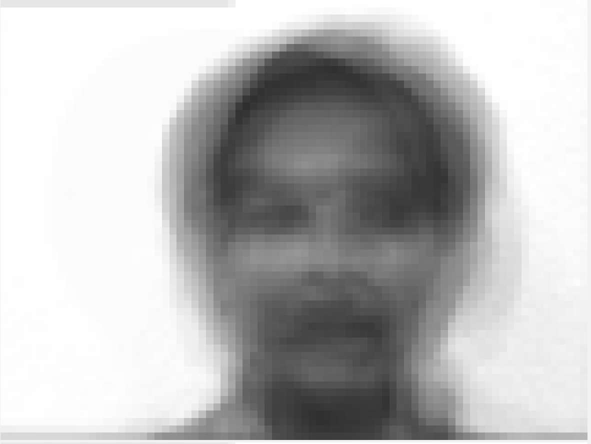
Two methods performed here that can control the effective complexity of a model are the ridge regression and the five cross-validation methods. The ridge regression allows for an alternate approach with smaller weights, making differences in complexity less impactful. The five-cross-validation performed allows for a weighted average to better predict the model's capacity to overfit.

Problem 4

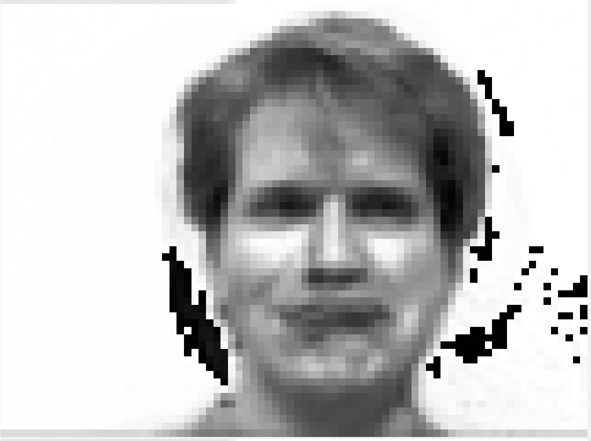
Eigenfaces of Different M values

Five representative images for each M =

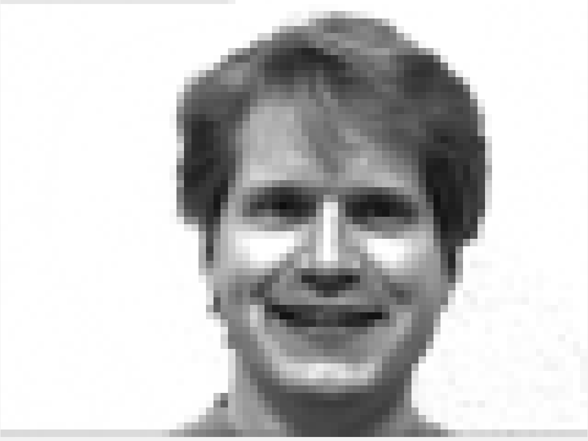
$$2, 10, 100, 1000, 4000$$



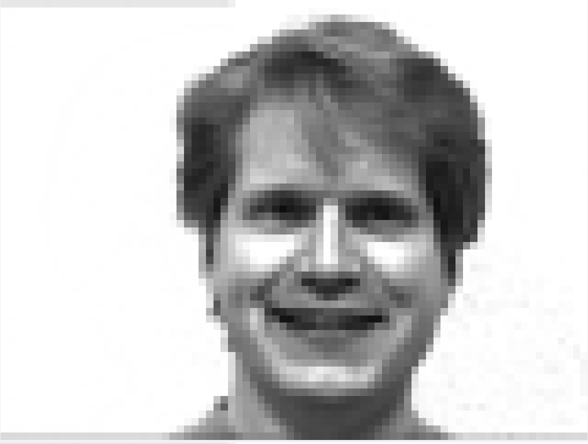
M = 2



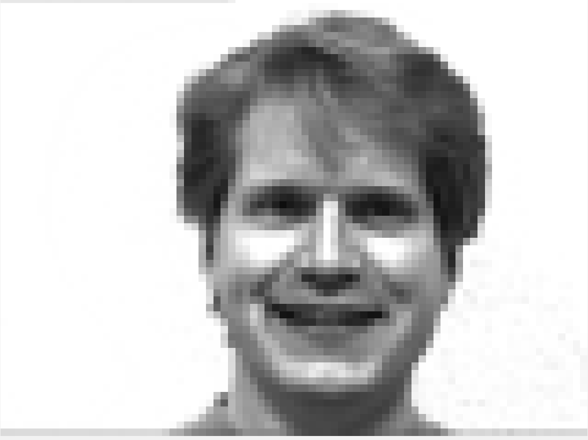
M = 10



M = 100

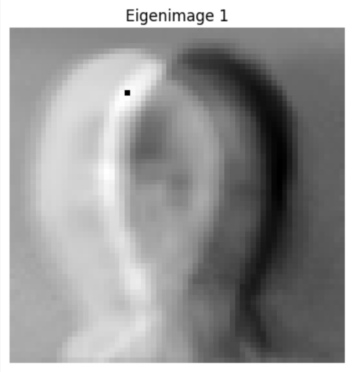


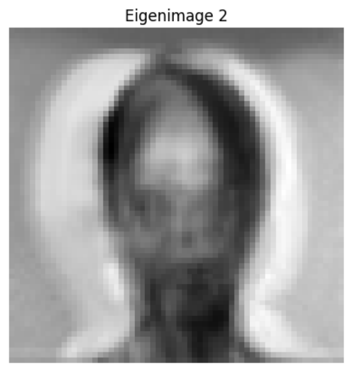
M = 1000

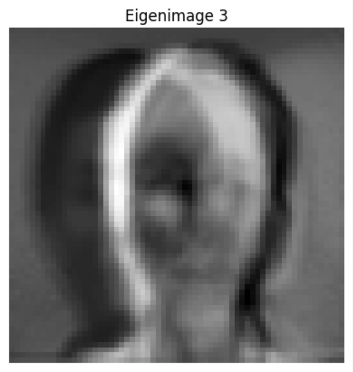


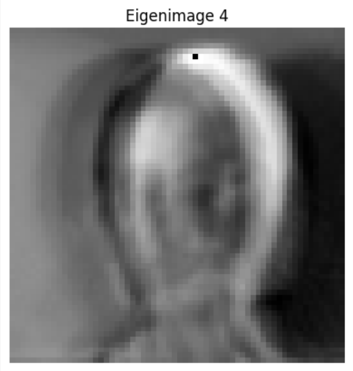
M = 4000

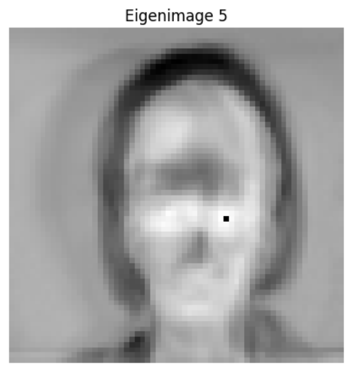
Eigenvector visualization

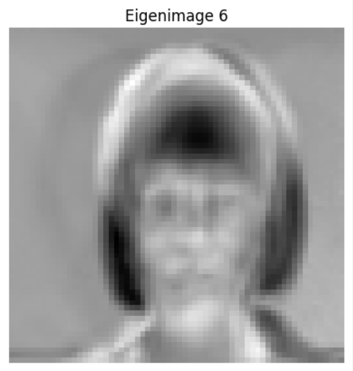


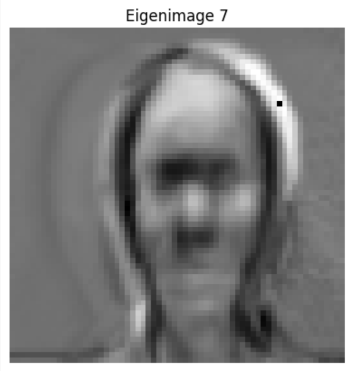


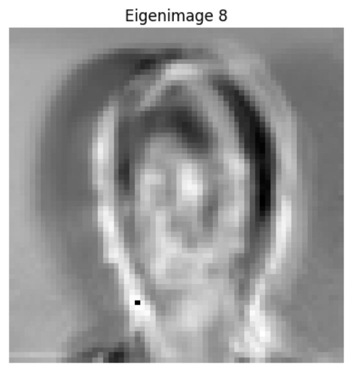


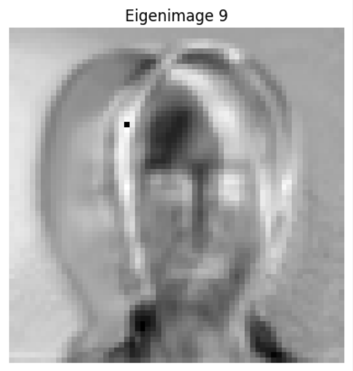


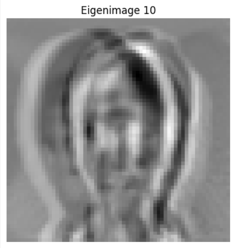








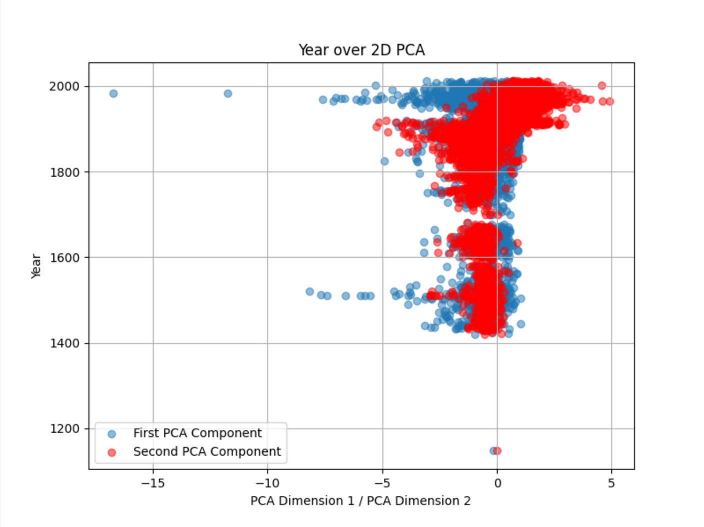




From viewing these images representing the 10 largest eigenvalues, I can see these images attempting to capture some of the features of the human face. The first four seem to capture the shape of the human head and outer features like hair and ears. Images 5-8 seem to capture inner features such as eyes, nose, and mouth, and the last three images (8-10) seem to be an overlap between facial features and the head shape of human faces.

Problem 5

5.1 Reproduced Plot



The advantage of a 2-D projection over a 1-D projection for year prediction is because a 2-D representation allows for another dimension to better separate data classes, as well as to better model relationships between features and the differentiation between these features.

5.3 Model results

Test MSE: 721206.3331215654
Most Accurate Prediction Index: 391
Actual Year: 1874.0, Predicted Year: 1874.1055335159572
Filename: 5291_italian-landscape-with-a-peasant-1874.jpg
Least Accurate Prediction Index: 690
Actual Year: 1984.0, Predicted Year: 18868.79263916057
Filename: 1803_untitled-holy-hole-1984.jpg