[redacted]

University of Illinois Fall 2024

ECE 364 Programming Methods for Machine Learning
Homework 4

Due on Thursday, October 10, 2024, 11:59pm on Gradescope

1. **[20 points] Solving Linear Regression in One Dimension**
Consider the following dataset:

$$\mathcal{D} = \{(-2,-4),(-1,-1),(0,-1),(1,0),(2,3)\},$$

where $(x_1, y_1) = (-2,-4), (x_2, y_2) = (-1,-1)$, and so on. We would like to perform linear regression according to the following optimization problem:

$$\min_{w_0,\ w_1} \frac{1}{2}\sum_{i=1}^{5}(y_i - w_1 x_i - w_0)^2.$$

Recall that the linear regression optimization program can also be written in vector format

$$\min_w J(w) = \min_w \frac{1}{2}\|\mathbf{X}^\top w - y\|_2^2,$$

where $w = \{w_1, w_0\}^\top$ contains the unknown variables. In general, this notation will work for linear regression with d variables for each observation where $\underline{\mathbf{X}^\top \in \mathbb{R}^{N\times d}}$.
With sufficient data, linear regression has the following closed-form solution w^*:

$$w^* = (\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}y. \tag{1}$$

(a) (5 points) Determine the $\mathbf{X}^\top$ matrix and vector y from the dataset $\mathcal{D}$.

Your answer:

$$X^T = \begin{bmatrix} -2 & 1 \\ -1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 2 & 1 \end{bmatrix} \qquad y = \begin{bmatrix} -4 \\ -1 \\ -1 \\ 0 \\ 3 \end{bmatrix}$$

(b) (5 points) Compute w^* for the given dataset using the closed-form solution of Eqn. 1. Please show your work.

Your answer:

$$XX^T = \begin{bmatrix} 10 & 0 \\ 0 & 5 \end{bmatrix} \qquad (XX^T)^{-1} = \begin{bmatrix} \frac{1}{10} & 0 \\ 0 & \frac{1}{5} \end{bmatrix}$$

$$w^* = (XX^T)^{-1}Xy = \begin{bmatrix} \frac{1}{10} & 0 \\ 0 & \frac{1}{5} \end{bmatrix}\begin{bmatrix} -2 & -1 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}\begin{bmatrix} -4 \\ -1 \\ -1 \\ 0 \\ 3 \end{bmatrix}$$

$$= \begin{bmatrix} \frac{3}{2} \\ -\frac{3}{5} \end{bmatrix}$$

(c) (4 points) Instead of utilizing the closed-form solution for linear regression, we may use gradient descent to iteratively solve for w. First, derive $\frac{\partial J}{\partial w}$.

Your answer:

$$\frac{\partial J}{\partial w} = XX^T w - XY$$

(d) (6 points) Let the initial point of $w^{(0)} = \{0, 0\}^\top$ and the step-size $\alpha = \frac{1}{20}$. Determine the next three iterates – $w^{(1)}, w^{(2)}, w^{(3)}$ – via gradient descent. Does $w^{(k)}$ seem to be approaching w^* as we iterate gradient descent?

Your answer:

$$w^{(k+1)} = w^{(k)} - \alpha \nabla J$$

$$w^{(1)} = w^{(0)} - \alpha \nabla J = \begin{bmatrix} 0 \\ 0 \end{bmatrix} - \frac{1}{20}\begin{bmatrix} 15 \\ -3 \end{bmatrix} = \begin{bmatrix} -\frac{3}{4} \\ \frac{3}{20} \end{bmatrix}$$

$$w^{(2)} = w^{(1)} - \alpha \nabla J = \begin{bmatrix} -\frac{3}{4} \\ \frac{3}{20} \end{bmatrix} - \frac{1}{20}\begin{bmatrix} -\frac{45}{2} \\ \frac{15}{4} \end{bmatrix} = \begin{bmatrix} \frac{3}{8} \\ -\frac{3}{80} \end{bmatrix}$$

$$w^{(3)} = w^{(2)} - \alpha \nabla J = \begin{bmatrix} \frac{15}{16} \\ \frac{-57}{320} \end{bmatrix}$$

Yes.

2. **[22 points] Harmonic Regression**

The linear regression optimization program for dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$, where $x_i \in \mathbb{R},\ y_i \in \mathbb{R}$, can be written as

$$\min_{w_1,\ w_0} \ \frac{1}{2}\sum_{i=1}^{N}(y_i - w_1 x_i - w_0)^2. \tag{2}$$

This model assumes $y_i \approx w_1 x_i + w_0$ and thus x_i is related to y_i by a first-order polynomial. We saw in class that we may use linear regression as well to approximate data as an n–th order polynomial, i.e. $y_i \approx w_n x_i^n + w_{n-1} x_i^{n-1} + \ldots + w_1 x_i + w_0$.

In this problem, we would like to consider another form of function approximation: harmonic regression. Consider the following model relating x_i and y_i:

$$y_i \approx A_1 \sin(\omega_0 x_i) + A_2 \sin(2\omega_0 x_i) + A_3 \sin(3\omega_0 x_i).$$

In this model, we assume that three harmonically related sinusoids with fundamental frequency $\omega_0 \in \mathbb{R}$ and unknown amplitudes (A_1, A_2, A_3) express the relationship $y_i \approx f(x_i)$, i.e. $f(x_i)$ is a weighted combination of three harmonic sinusoids.

(a) (4 points) State the optimization program for the above harmonic regression model like we did above for linear regression above in Eqn. 2.

Your answer:

$$\min_{A_1 A_2 A_3} \frac{1}{2}\sum_{i=1}^{N} \left(y_i - A_1 \sin(w_0 x_i) - A_2 \sin(2 w_0 x_i) - A_3 \sin(3 w_0 x_i)\right)^2$$

$$Y - \sin w. \begin{bmatrix} A_2 \\ A_4 \\ A \end{bmatrix}$$

(b) (5 points) When performing polynomial regression, we saw in lecture that a transformation of the data denoted as $\Phi(\mathbf{X})^\top := \mathbf{\Phi}^\top \in \mathbb{R}^{N\times d}$ could be used to find the solution for polynomial regression where the solution w^* was given by $w^* = (\mathbf{\Phi}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}y$. For example, we could write $\mathbf{\Phi}$ as the following for quadratic regression:

$$\mathbf{\Phi}^\top = \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_N^2 & x_N & 1 \end{bmatrix}.$$

We may use the same approach now for harmonic regression such that the solution A^* may be found by:

$$A^* = (\mathbf{\Phi}\mathbf{\Phi}^\top)^{-1}\mathbf{\Phi}y. \tag{3}$$

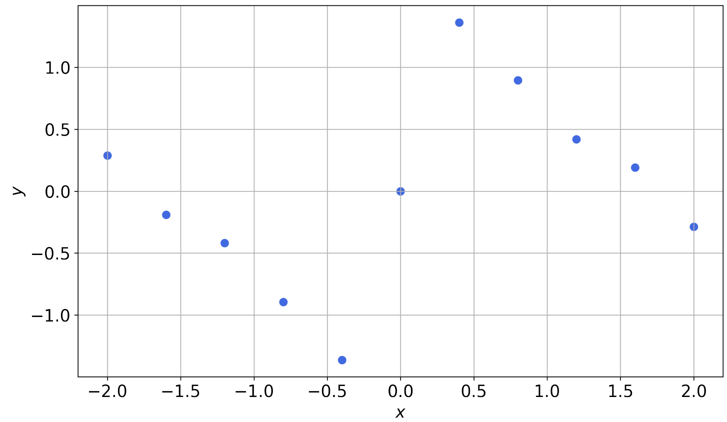
Give $\mathbf{\Phi}^\top$ for the given harmonic regression problem for fundamental frequency ω_0. Assume we have $N = 5$ data points. **Hint:** Your $\mathbf{\Phi}^\top$ should be of shape $(5, 3)$.

Your answer:

$$\min \frac{1}{2}\sum_{i=1}^{N} \left| Y - \begin{bmatrix} \sin w_0 x_0 & \sin 2w_0 x_0 & \sin 3w_0 x_0 \\ \sin w_0 x_1 & \vdots & \vdots \\ \vdots & & \\ \sin w_0 x_N & \sin 2w_0 x_N & \sin 3w_0 x_N \end{bmatrix} \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix} \right|^2$$

$$\Rightarrow \phi^T = \begin{bmatrix} \sin w_0 x_0 & \sin 2w_0 x_0 & \sin 3w_0 x_0 \\ \sin w_0 x_1 & \sin 2w_0 x_1 & \sin 3w_0 x_1 \\ \sin w_0 x_2 & \sin 2w_0 x_2 & \sin 3w_0 x_2 \\ \sin w_0 x_3 & \sin 2w_0 x_3 & \sin 3w_0 x_3 \\ \sin w_0 x_4 & \sin 2w_0 x_4 & \sin 3w_0 x_4 \end{bmatrix}$$

(c) (10 points) On the course website, we have provided a `hw4_p2.zip` file. If you download and unzip this file, you will find ".npy" data files and a Jupyter notebook (.ipynb) file. The data contains 11 (x, y) coordinates to attempt this harmonic regression problem. The given data will generate the below scatter plot.



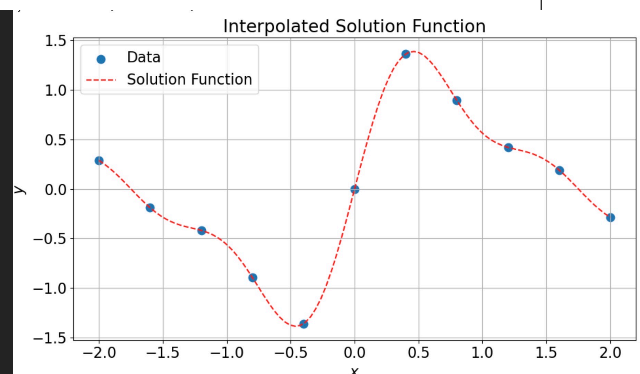
Given that $\omega_0 = \frac{4\pi}{7}$, implement the harmonic regression solution from Eqn. 3 using the provided Jupyter notebook. **State your solved values for A_1, A_2, and A_3 and share the interpolated function plot of your solution (final plotting result of the last code cell).** **Hint**: Your solution should perfectly match the data with no errors, i.e. the sum of squared errors is zero and A_1, A_2, and A_3 are intentionally "nice" values.

Your answer:

```python
def build_phi(x):
    omega = 4*np.pi/7
    # fill in this function to return the phi matrix
    phi_tranpose= np.column_stack((np.sin(omega * x), np.sin(2 * omega * x), np.sin(3 * omega * x)))
    return np.transpose(phi_tranpose)

def harmonic_regression_solution(phi, y):
    # implement the closed-form harmonic regression solution to return the three amplitude values "A"
    A = np.linalg.inv(phi @ np.transpose(phi)) @ phi @ y
    return A

def evaluate_solution(A, t):
    # helper function for visualization
    omega = 4*np.pi/7
    return A[0]*np.sin(omega*t)+A[1]*np.sin(2*omega*t)+A[2]*np.sin(3*omega*t)

phi = build_phi(x)
# A_star = harmonic_regression_solution(phi, y) # uncomment this when you are ready to test your implementation
A=harmonic_regression_solution(phi,y) # dummy solution, delete this when testing your implementation
t = np.linspace(-2, 2, 100) # points for visualizing your solution
f = evaluate_solution(A, t)

print('My A solution is: {}'.format(A)) # report these values

# visualize solution
plt.figure(figsize=(10, 6))
plt.scatter(x, y, s=80, label='Data')
plt.plot(t, f, color='red', linestyle='dashed', label='Solution Function')
plt.legend()
plt.xlabel(r'$x$')
plt.ylabel(r'$y$')
plt.title('Interpolated Solution Function')
plt.grid(True)
plt.tight_layout()
```

✓ 0.2s Python

```
My A solution is: [1.   0.5  0.25]
```



$A_1 = 1$

$A_2 = \frac{1}{2}$

$A_3 = \frac{1}{4}$

(d) (3 points) What is the minimum number of points N such that we may find a unique solution to this particular harmonic regression problem? Explain your reasoning.

Your answer:

we need at at least 3 points because there are 3 unknown parameters, to solve them, we need at least 3 distinct points which are all on the curve

3. **[8 points] Linear Regression in Higher Dimensions**

   Consider a problem setting where we would like to apply linear regression to images of people in order to predict their age. More specifically, for dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, each (x_i, y_i) pair is a 50×50 pixel grayscale image for x_i and the person's age for y_i.

   (a) (4 points) Assume that we vectorize each image, i.e. we flatten each 50×50 image into a one-dimensional vector with the same number of pixels, and that we have 10 image examples. What would be the dimensions of $\mathbf{X}^{\top}$ and y with respect to the closed-form solution in Eqn. 1? What would be the dimensions of the solution vector w^*?

   Your answer:

   the dimension of X^T is $50 \times 50 \times 10 = 25000$

   the dimension of Y is $10 \times 1 = 10$

   the dimension of w^* is $2500 \times 1 = 2500$

   (b) (4 points) Is it possible to obtain the closed-form solution from Eqn. 1? If not, what could we attempt to obtain a solution? Explain your answer.

   Your answer:

   No because $(X^T X)$ is a singular matrix and regularization may help to make it invertible