

For Instructor/TA Use Only

| | |
|---------------------------|--|
| Student Name: | |
| Student Last Name: | |
| Student Andrew ID: | |

| | | |
|------------------------------|------------------------------|------------------------------------|
| Q1 (max 50) | Q2 (max 50) | Q3-extra (max 20) |
| | | |

24-787 Artificial Intelligence and Machine Learning for Engineering Design

Homework 6

Regression

Naming and Folder structure: Name a root folder "andrewid_hw#". Example: lkara_hw1. In that root folder, create subfolders such as q1, q2, q3 etc. each corresponding to a question in the assignment. The files and documents related to each question should go into the corresponding subfolder.

File types: Your folders and subfolders should not contain any files except .pdf, .doc, .docx, .m, .mat, .txt .zip files. If you want to take photos of your assignment, just make sure that combine all the jpg into a single pdf file and make it clear. Unorganized and illegible files will be penalized. Take care in arranging your illustrations, written solutions, photos. If any, do not scan the your hand-written solutions in the highest resolution.

What to include: Only submit the required files that you create or modify. For programming questions, include your source code (.m and/or .mat file rather than text) in your submission. All other files, including the ones we provide as supporting functionality, are unnecessary because we already have copies of them. Unless we tell you otherwise, make sure that you only submit the file you edited/changed. For .m files, we expect to click the "run" button and everything should work (obviously, after we include the additional support files and data that was given to you with the assignment).

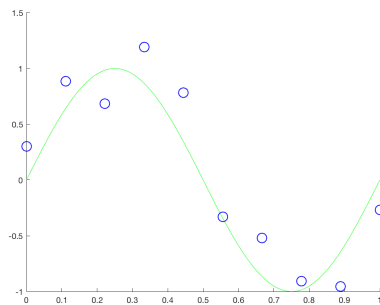
Readme files: Provide a readme.txt within each questions' subfolders when necessary to aid grading. This is useful when there are many files of the same type. If your folder structure and your file names are logical, you should not need a readme.txt file.

Submission: Zip the entire assignment by compressing the root folder. Example: lkara_hw1.zip. Do not use .rar. Make sure your entire submission file is less than 20MB when zipped. You shall use the online homework submission platform. The submission entry can be found at the top right of homework/assignment page. Make sure your submission is in .zip format, otherwise it cannot be uploaded. Feel free to re-submit your homework before deadline if you find any mistake in your previous submission. Only the last submission will be graded.

1. (50 points)

In this problem, you will implement a linear regression model. There are three datasets, with 10, 15, and 100 points. In each data file, the first column is the input and the second column is the output. The underlying ground truth function is $y = \sin(2\pi x)$.

- (a) Plot the three datasets side by side using blue circles for the data points. **Show these plots on a report by taking screenshots similar to the way your are seeing in this assignment. On a given plot, plot the ground truth function as a green line using $x = \text{linspace}(0, 1, 100)'$ as your input vector.** For example, data10 should look like this:

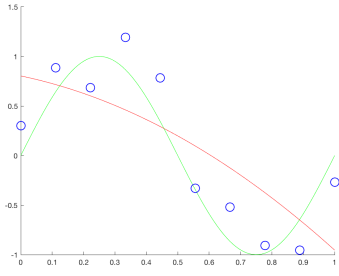


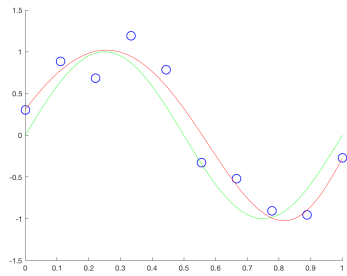
- (b) Edit and **submit file q1.m**. Write a polynomial regression function with $L2$ regularization using the formulation involving normal equations and matrix inversion to compute the regression coefficients. You cannot use Matlab's in-built regression functions. Your regression function should be of the form:

$$y = c_m * x^m + c_{m-1} * x^{m-1} + \dots + c_0 * x^0$$

You will fit models for four different m values from the set $\{1, 2, 6, 9\}$. Arrange your coefficient vector \mathbf{c} such that its first element is the coefficient for the largest power of x (i.e., c_m) and its last element is the coefficient for the smallest power of x (i.e., c_0).

Perform your regularized regression on the three data sets, where the function $(Ac - b)'(Ac - b) + \text{lambda} * c' * c$ is to be minimized. Using your regression model, estimate the output $y_{\text{estimated}}$. Plot the estimated function as a red line. Also, for each case output the coefficient vector c . In Matlab, use *format bank* to output the values of c . **You can report your results by populating the cells of the following tables (one example cell per table is provided).**

| lambda=0 | m=1 | m=2 | m=6 | m=9 |
|----------|-----|---|-----|-----|
| data10 | |  <p>$c = [-1.09; -0.66; 0.80]$</p> | | |
| data15 | | | | |
| data100 | | | | |

| lambda=exp(-10) | m=1 | m=2 | m=6 | m=9 |
|-----------------|-----|-----|-----|--|
| data10 | | | |  <p>$c = [-3.53; -1.85; 2.30; 6.47; 6.81; 0.25; -9.31; -6.83; 5.10; 0.31]$</p> |
| data15 | | | | |
| data100 | | | | |

- (c) In a few sentences, **explain** your observations for *data10*, and $m=9$ (for both $\lambda = 0$ and $\lambda = \exp(-10)$). Also **explain** your observations for *data100*, and $m=9$ (for both $\lambda = 0$ and $\lambda = \exp(-10)$). When data is scarce, do you recommend using high or low degree polynomials for regression, why? How does regularization help?

- (d) The root-mean-square error (E_{RMS}) is one way of computing the quality of your fit. E_{RMS} is defined as:

$$E_{RMS} = \sqrt{2.0 * E(c) / N} \text{ where } E(c) = \|y_{\text{groundtruth}} - y_{\text{estimated}}\|^2 \text{ and } N \text{ is the length of } y_{\text{estimated}} \text{ (or } y_{\text{groundtruth}}).$$

For *data10* and $\lambda = 0$ (no regularization), **compute and output the E_{RMS} between:**

- i. the ground truth function and your estimation for the four different values of m .
- ii. the data points and your estimation for the four different values of m .

Overlay and plot your results as a function of m with (i) as green circles and (ii) as blue circles. Note that in reality, one does not have access to the ground truth function. Hence, the E_{RMS} error has to be computed between the sampled data points and the estimation. **In (ii), what problems do you foresee?** Hint: In (ii), for $m=9$, E_{RMS} is zero, but does this mean this is a good regression model?

- (e) In part (d), using (i), **what is the best value of m ? Why?**

- (f) When the size of vector c is large, the normal equations may be expensive to solve using matrix inversion. Instead, one can use gradient descent to compute c . Derive the gradient descent approach for L2 regularized linear regression. Specifically:

$$E(c) = (Ac - b)^T (Ac - b) + \lambda c^T c$$

Find $dE(c)/dc = 0$ using the iterative update rule:

$$c^{new} = c^{current} - \alpha * dE(c)/dc$$

What is $dE(c)/dc$? Derive this gradient on paper. Hint: We have shown this derivation in class. Use the rules of matrix/vector differentiation:

$$\begin{aligned} \frac{\partial}{\partial \mathbf{x}} [\mathbf{M}\mathbf{x}] &= \mathbf{M} \\ \frac{\partial}{\partial \mathbf{x}} [\mathbf{y}^T \mathbf{x}] &= \frac{\partial}{\partial \mathbf{x}} [\mathbf{x}^T \mathbf{y}] = \mathbf{y} \\ \frac{\partial}{\partial \mathbf{x}} [\mathbf{x}^T \mathbf{M}\mathbf{x}] &= [\mathbf{M} + \mathbf{M}^T] \mathbf{x} \end{aligned}$$

- (g) In this problem, could gradient descent get stuck in a local minimum? **Explain why / why not?**

- (h) **Edit and submit file *q1_gd.m*.** Using the following settings:

- *data100*
- $\lambda = \exp(-10)$
- $m=9$
- c initially set to the zero vector
- $\alpha = 0.01$ (damping factor).
- Number of iterations = 10,000

Compute c using gradient descent and **output the result on paper**. No need to normalize the data in this problem. **Also output on paper the difference vector between the c your just computed and the one you computed in part (b).** That is, output:

$$c_{\text{diff}} = c_{\text{part_h}} - c_{\text{part_b}}$$

- (i) For part (h), **explain what happens when $\alpha = 0.9$ and why.**

2. (50 points)

Imagine a 3D temperature field $T(x,y,z)$. You do not have access to this function directly, but you have made several temperature measurements:

$T(0,0,0)=10$ Celsius
 $T(8,6,1)=15$ Celsius
 $T(5,2,8)=20$ Celsius
 $T(8,2,6)=22$ Celsius
 $T(5,1,2)=16$ Celsius
 $T(3,3,3)=23$ Celsius
 $T(9,8,2)=18$ Celsius
 $T(3,6,5)=19$ Celsius
 $T(4,6,9)=25$ Celsius
 $T(1,8,2)=20$ Celsius
 $T(1,1,2)=28$ Celsius
 $T(6,4,2)=27$ Celsius

You would like to determine the temperature $T@ (5,5,5)$.

- Derive the A, c, and B matrices for a linear least squares using a **linear** function. **Write and submit the corresponding Matlab code **q2.m** to determine $T(5,5,5)$. Report $T(5,5,5)$ on paper.** You may use any technique/function you would like. **Feel free to normalize the data and apply L2 regularization, but that is optional and will not be graded.**
- Based on (a), at point $(5,5,5)$, in which direction should one move to experience the largest decrease in temperature in the immediate neighborhood of the point? **Report this direction on paper together with your derivations. You can use Matlab to compute the necessary numbers.**

3. (20 points- **extra**)

Edit and submit file **q3.m Use Matlab's **quagprog** to solve Problem 1, part h. Show your derivation of the formulation for **quagprog** on paper.** Note that the objective function to minimize is $E(c)$. Also note that you will not need *alpha*, or the number of iterations.