

Name: _____

CSE 3521

Artificial Intelligence

SP'21

Homework Assignment #9 (23 points)

This assignment requires coding in Javascript. Use the template provided in `HW10_template.zip` to get started. See `param_est.js` for the functions you need to complete.

Use the data points provided in the template for all of the following problems.

Linear Least Squares

1. Complete the function `calc_linLSQ_line()`: Use linear least squares to estimate the parameters (a and b) for the following model:

$$f(x, a, b) = a x + b$$

Calculate the sum of squared error using your results. Also, create a plot of the model function using the parameter values you found (the “Save As Img” button can be used for this). Include these in your report. (4 pts)

2. Similar to problem 1, complete the function `calc_linLSQ_poly()`: Use linear least squares to estimate the parameters for various polynomials, such as:

$$\begin{aligned} f(x, a, b, c) &= a x^2 + b x + c \\ f(x, a, b, c, d) &= a x^3 + b x^2 + c x + d \\ f(x, a, b, c, d, e) &= a x^4 + b x^3 + c x^2 + d x + e \end{aligned}$$

Create a plot and calculate SSE as before, for your report. Compare the plots and errors to your results from problem 1. Which of these 4 models (including problem 1) do you think best fits the data? Why? (7 pts)

Non-linear Least Squares

3. Derive the function for the Jacobian matrix for the following non-linear model:

$$f(x, \mathbf{p}) = a x^b + c x + d$$

In addition to including this derivation in your report, implement this in the `calc_jacobian()` function. (2 pts)

Hint: $\frac{d}{dt} c^{a t} = a c^{a t} \log c$

4. Complete the function `calc_nonlinLSQ_gaussnewton()`: Use Gauss-Newton non-linear least squares to estimate the parameters for the function from (3).

Use the following for your initial guess:

`a=0.5, b=2, c=0.5, d=0.5`

Stop after 10 iterations. (Note, these are the default values in the template.)

As with problems 1-2, create a plot of the resulting model function. Also, calculate the sum of squared error after each iteration. Do you think this model fits the data better than previous ones? Why or why not? (5 pts)

5. Complete the function `calc_nonlinLSQ_gradient descent()`: Use Gradient Descent non-linear least squares to estimate the parameters for the function from (3).

Use the same initial guess as (4). Use a learning rate of 0.001 and stop after 5000 iterations.

As with problem 4, create a plot and find the sum of squared error after each iteration. How does this algorithm compare to (4)?

Try different values for learning rate. Can you achieve a better convergence rate (lower error or less iterations for same error)? What are your observations on how the algorithm behaves with different values? (5 pts)

Create (and submit in class) a report including answers to the asked questions and a printout of your code. Also, **create a ZIP archive of your code files** and submit it in the Homework 10 dropbox on Carmen.

Note: This assignment may be completed as a group of up to two people. Each group should submit a single report (with both names on it) and only one group member should submit the combined code to Carmen.

Tips:

If you need to print out debug statements, you may use the `console.log()` function to print out to the browser's debug console. To access this log, use Ctrl-Shift-J in Chrome or Shift-F5 (Console tab) in Firefox.

Alternately, you can use the `helper_log_write()` function (in `param_est_helper.js`) to output to the log region on the web page.