GEO5017 A1
Linear Regression

Question 1. Basic model (20%)

Suppose we have observed N data points $\{x_i, t_i\}_{i=1...N}$, answer the following questions:

1) What do we need to do with the data before we can formulate a linear regression problem? Use one short sentence to answer this question. (10%)

2) What would be the optimal solution given your hypothesized model? Use one short sentence to answer this question. (10%)

Question 2. Programming: drone (60%)

We are testing a tracking program, and we evaluate it with the help of a drone. The tracking program estimates its global position concerning the drone's initial position.

1) The tracker yields these tracked positions at a frequency of $1Hz$

$$\mathcal{P} = \left\{ \begin{pmatrix} 2 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} 1.08 \\ 1.68 \\ 2.38 \end{pmatrix} \begin{pmatrix} -0.83 \\ 1.82 \\ 2.49 \end{pmatrix} \begin{pmatrix} -1.97 \\ 0.28 \\ 2.15 \end{pmatrix} \begin{pmatrix} -1.31 \\ -1.51 \\ 2.59 \end{pmatrix} \begin{pmatrix} 0.57 \\ -1.91 \\ 4.32 \end{pmatrix} \right\}$$

   at the corresponding times $\mathcal{T} = \{1, 2, 3, 4, 5, 6\}$. Plot the trajectory through these data points with your tool of choice. (10%)

2) Use the Polynomial Regression method introduced in the lecture to answer the following questions and provide details in your report. You can choose your favorite programming language to formulate and solve the regression problems (10%). You **must** use your own implementation of the Gradient Descent solver (10%).

   (a) Assume the drone flies with a constant speed in each of the three dimensions, what speed does it have? What is the residual error (i.e., sum-of-squares error) of the estimated positions? (10%)

   (b) Assume the drone flies with a constant acceleration in each of the three dimensions, what is the residual error now? Is the error higher or lower? Why, and can you still improve your model? (10%)

   (c) According to your 'constant acceleration' model, what is the drone's most likely position in the next second (i.e., $t = 7$)? Plot this position together with previous positions. (10%)

Submission

Please compress all the following into a single archive titled **GEO5017_A1_Group_XX.zip** (where "XX" is your group ID. In case you have an updated version, please append "_V2" to its title) and submit it to BrightSpace:

- **Report** (max 2 pages excluding figures and tables) (20%)
  - The report must contain the answers and necessary explanations. Please focus on the mathematical formulations and the analysis of your models, and try to avoid unnecessary details like how the data is stored in your code and how a curve is plotted.
  - The report must describe how *learning_rate* was chosen and the converge/stop conditions of your gradient descent solver.
  - The report must contain a short description of who did what.

  To ensure formal scientific writing, the following rules apply for assessment:

  - Any misunderstanding or misconception of main concepts: 10% deduction.
  - Multiple unclear or ambiguous descriptions: 10% deduction.
  - Mulitple typos, grammar issues, format issues (e.g., a figure/table without a caption or multiple figures/tables with the exact same caption, unindexed or unreferenced figures/tables), consistency issues (e.g., upper case and lower case used interchangeably, normal font and italic font used interchangeably, misuse of symbols in notations): 10% deduction.
  - The report is more than half a page over length: 10% deduction.
- **Source code** (for Question 2)
  - The source code, archived in a 'code' subfolder. The code should build, run, and reproduce your results without changes.
    - In your implementation, please make *learning_rate* and *num_iterations* (or another preferred converge condition) tunable, to allow the teacher(s) to further check your implementation when necessary.
    - For Python code, **only** '*.py' files are accepted. If you use Jupyter Notebook for development, make sure you submit '*.py' files ('*.ipynb' files will not be accepted). For C++ code, **only** '*.cpp' files (together with a 'CMakeLists.txt' file) are accepted.
    - If you have multiple source code files that should run in a specific order, **do** name them in a way such that they are well ordered (e.g., '1_data_fitting.py', '2_solver.py', '3_visualization.py'). If no such information is provided, we will **expect a 'main.py' file** as the only entry point.
    - Provide a 'ReadMe.txt' file to briefly explain how to run the code and reproduce the results, e.g., dependence on external libraries/packages, the path to data, and where to find the results in case you save results or figures into files.
    - It is recommended to test your code on different operating systems, as your code may be evaluated on an OS different from that it was developed.

  - [*optional*] Provide a link to the GitHub repository (if you use GitHub) in the 'Experiment' section of your report. You are encouraged to collaborate with your teammates on GitHub.

Note: The report should be as concise as possible (within 2 pages excluding figures, tables, and references), but it should provide sufficient information to re-implement your methods and reproduce your results. Try to use the mathematical language (i.e., equations) as much as possible. You will have to omit less important details and discussions. Irrelevant descriptions and discussion may lead to the deduction of points.