CS F320 – Foundations of Data Science

Assignment – 3

Submission deadline: TBD

General Instructions:

- This assignment is a coding project and is expected to done in groups. Each group can contain at most three members. Make sure that all members in the group are registered to this course and please try to maintain the same group for all the assignments.
- This assignment is expected to be done in Python using standard libraries like NumPy, Pandas and Matplotlib. You can use Jupyter Notebook. For this specific assignment, you are allowed to use the `sklearn.preprocessing.PolynomialFeatures` class to generate the polynomial features for the data. Note that, no other class or ML libraries/APIs must be used.
- Refrain from directly copying codes/snippets from other groups or the internet as all codes will be put through a plagiarism check.
- All deliverable items (ex. `.py` files, `.ipynb` files, reports, images) should be put together in a single `.zip` file. Rename this file as `A2_<id-of-first-member>_<id-of-second-member>_<id-of-third-member>` before submission.
- Submit the zip file on CMS on or before the aforementioned deadline. Please note that this is a hard deadline and no extensions/exemptions will be given. The demos for this assignment will be held on a later date which shall be conveyed to you by the IC. All group members are expected to be present during the demo.

Problem Statement:

- In this assignment, you will be implementing Polynomial regression (with degrees varying from 1 2, . ., 10) using Gradient Descent and Stochastic Gradient Descent backed up by Lasso and Ridge regularization. But before implementing the algorithms, you are expected to pre-process your data which includes shuffling the data, standardizing/normalizing the values and creating a random 70-20-10 split to aid in training, validation and testing respectively. Vectorize your algorithms as much as possible to efficiently carry out the computations. Try to print the error value after every 50 iterations during training for better visualization.
- The dataset consists of three features i.e. age, bmi and number of children of an individual. Drop the 'number of children' feature before pre-processing the data. Using the remaining features (age and bmi), you are expected to predict the insurance amount for that person by constructing matured polynomial features (obtained from the `PolynomialFeatures` class) and optimizing the weights by using GD/SGD. Try to write a clean, modularized and vectorized code which can solve the above problem. Please refrain from hardcoding any part of your code, until unless it is absolutely necessary.

What needs to be documented in your report:

- Firstly, give a brief description of your model, algorithms and how you implemented the regularization.
- Secondly, tabulate the minimum training and testing error achieved by your model by using polynomials of degree 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 to predict the output. Now add a regularization term and use random values uniformly chosen from the range [0, 1] for the regularization parameter. Tabulate the minimum and training, validation and testing error you get by incorporating regularization.

- Finally, visualize the surface plots of your predictions (using matplotlib and Axes3D) that you obtained by using polynomials of varying degree and comment on how overfitting actually works.

Questions to ponder on:

- What happens to the training and testing error as polynomials of higher degree are used for prediction?
- Does a single global minimum exist for Polynomial Regression as well? If yes, justify.
- Which form of regularization curbs overfitting better in your case? Can you think of a case when Lasso regularization works better than Ridge?
- How does the regularization parameter affect the regularization process and weights? What would happen if a higher value for λ (> 2) was chosen?
- Regularization is necessary when you have a large number of features but limited training instances. Do you agree with this statement?
- If you are provided with D original features and are asked to generate new matured features of degree N, how many such new matured features will you be able to generate? Answer in terms of N and D.
- What is bias-variance trade off and how does it relate to overfitting and regularization.

Link for the dataset:
https://drive.google.com/file/d/1plkY18tLL4P0DLlUwYB6l1P4YH24_lOQ/view?usp=sharing