Homework 01

Complete this assignment as an electronic notebook, saved as a Jupyter .ipynb file. You can use colab or desktop python. To submit your assignment, please commit your .ipynb notebook file, as well as a .pdf printout of the file, to your repository. These should be pushed to GitHub, and also shared with user artkuo.

1. Run the `gradientdescent.ipynb` notebook, which demonstrates gradient descent and linear regression. Answer a few questions (please answer with full sentences):
   a. The `LinearRegression` function returns a least-squares solution using pseudo-inverse. This is fast and accurate. In what machine learning situations might it be helpful to use gradient descent?
   b. Briefly describe how the (1.) gradient descent stops, using the precision variable.
   c. Try modifying the (2.) multivariable gradient descent to stop when the norm of the gradient (look up numpy's norm function) is below some threshold, instead of using precision. Demonstrate stopping below 1e-6.
   d. Demonstrate the effects of choosing too high of a learning rate, and too low of a learning rate.
   e. Comment on the path traced by the weight estimates as gradient descent progresses. Does each iteration move in the direction of steepest descent?
   f. Briefly distinguish between Newton's Method and Gradient Descent. How are they different?

2. Write and demonstrate your own stochastic gradient descent function. This should be very similar to the `gradient_descent1` provided. The difference is that instead of using the entire set of data, each iteration only uses the latest sample of x and y.
   a. Run 100 iterations of stochastic gradient descent, and compare the progress against 100 iterations of gradient descent. Comment on performance.
   b. Once stochastic gradient descent has run through all of your data, that is called one epoch. Multiple epochs are usually needed to converge on a good solution. (It may help to use the remainder operator, e.g. `i % N`.) Modify your function accordingly, and demonstrate performance against regular gradient descent after 1000 iterations (10 epochs).
   c. Plot the path traced by the weight estimates with stochastic gradient descent. Does each iteration move in the direction of steepest descent? When the estimate is close to correct, does the algorithm converge toward a single value?
   d. Demonstrate what happens with stochastic gradient descent if the learning rate is too low or too high.

3. Consider generalizing your function for multivariate regression, where y may be a vector of outputs. Briefly describe what changes would be needed to work with multivariate y and multivariate x.