Princess Tara Zamani
Adv Opt for Machine Learning
Homework 9
Stochastic Gradient Descent Implementation

The objective of this assignment was to apply stochastic gradient descent to solve the least-mean-square problem defined as:

$$min_{x\in R^n} f(x) = \frac{1}{2m}\|Ax - b\|_2^2$$, where $x \in R^n$, $A \in R^{m\times n}$, $b \in R^m$.

The stochastic gradient descent is given below. At iteration k, randomly sample B data points (named B_k) from $\{a_i, b_i\}_{i=1}^m$.

$$x_{k+1} = x_k - \eta\frac{1}{B}\sum_{i\in B_k}(a_i^T x - b_i)a_i$$

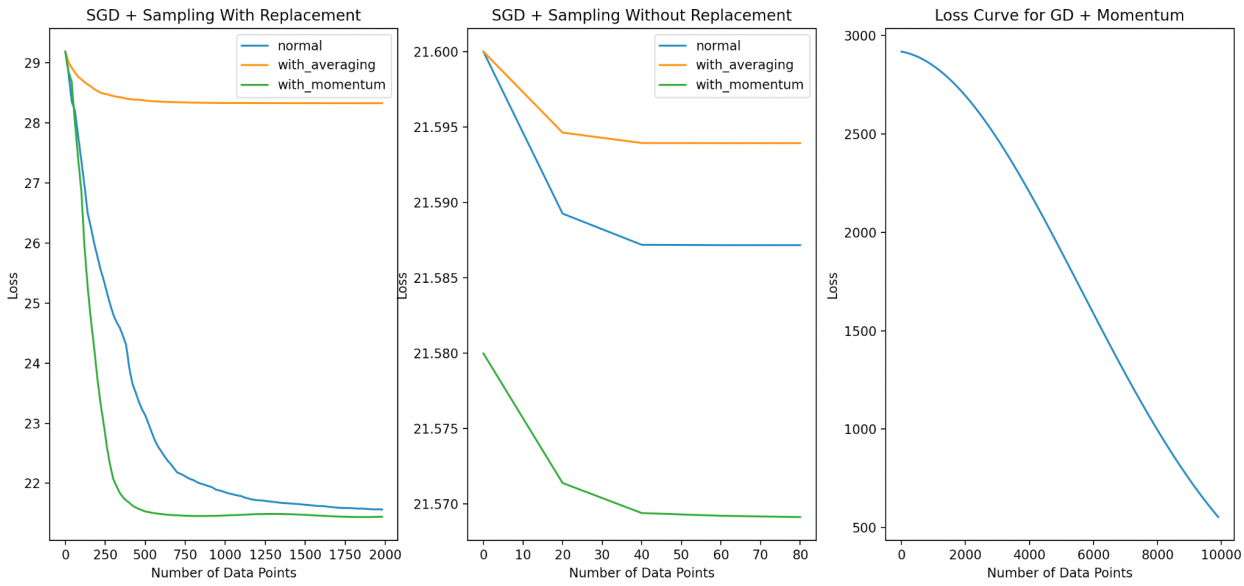
The underlying signal (z), and the initialization of x_0 were generated randomly using the numpy.random.randn function. The following initial parameters were set: variance = 0.1, n = 1000, m = 100, learning rate = 1e-3, batch size = 20, and k_iterations = 1000.

Simulation 1

The following algorithms were implemented to compare the impact of sampling methods:

- SGD + sampling with replacement (every Bk is draw from {ai,bi}mi=1)
- SGD + sampling with replacement + model averaging
- SGD + sampling with replacement + momentum
- SGD + sampling without replacement (reshuffle {ai,bi}mi=1 at the beginning of every epoch)
- SGD + sampling without replacement + model averaging
- SGD + sampling without replacement + momentum
- GD(gradient descent) + momentum

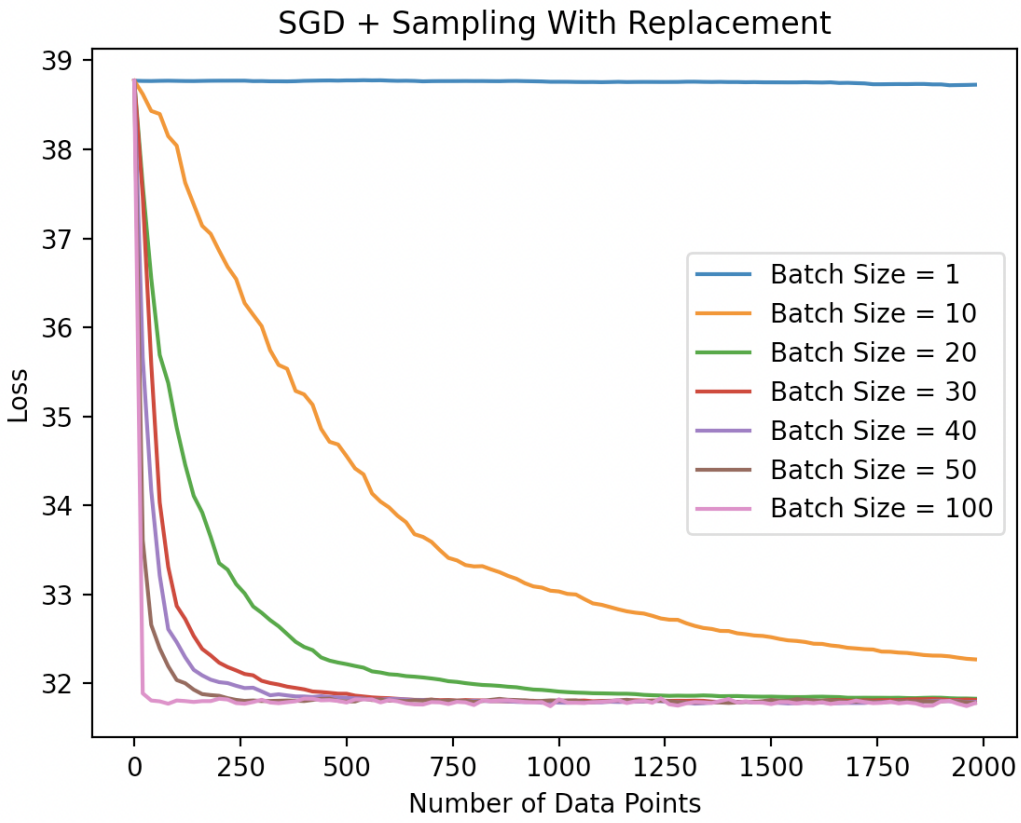
The results of the objective function vs number of data points for all algorithms are shown below.



The plots were created in the following groups: SGD with sampling replacement, SGD without sampling replacement, and GD with momentum. The number of data points were determined by multiplying the batch size with the number of iterations per method. Since there is no epochs in SGD with replacement, this would just be k_iterations * batchSize = 100*20. With the presence of epochs in SGD without replacements, each epoch had up to 5 iterations, therefore, the number of points 4*20. Lastly, the regular GD has m samples per iteration, hence 100*100.
Comparatively, we see that SGD without replacement takes far fewer data points to converge compared to the other methods. Within this group, the SGD + sampling without replacement + momentum was the fastest method to converge overall.

Simulation 2

The objective of this simulation was to use the SGD + sampling with replacement method and try varying the batch size between 1 and m. There is a linear relationship between batch size, B, and learning rate, $learning\ rate\ =\ \alpha\ *\ B$, where alpha is a tuned parameter. Alpha was set to 1e-4 to obtain the following results.



As the batch size increased, the convergence got faster. Within these results, the fastest convergence occurred when the batch size was equal to 100.