1. Partitioning of the dataset into training and validation portions must be done randomly to maintain the probability distribution of the original sample. That is the set of all possible data D is defined by some probability distribution P(y | x). The dataset provided is hopefully selected uniformly at random from D, implying that the dataset should have approximately the same P(y | x) as D. When we further partition the dataset into two pieces, we want to ensure, to the best of our ability, that we again maintain the same probability distribution in both of the partitions. The way to accomplish this is to select from the dataset uniformly at random.
2. Plot
   1. Do you see any pattern in the graph?
      1. Linear:
         1. As expected due to the convergence proof in class, the training error of perceptrons on the linear dataset always converges to a single value and then remains constant no matter how many iterations are performed after the convergence point.
      2. NonLinear:
   2. Do you see any trends in each plot? Do you see any relation between the two plots?
      1. Linear:
         1. For both the training and validation error plots, the error trends downward.
      2. NonLinear:
         1. Yes
      3. Linear:
         1. Yes
      4. NonLinear
         1. Yes
3. Learning Rate
   1. How does varying the learning rate affect the graph? Do they have any effect on how fast/slow your perceptron converges?
      1. Linear:
      2. NonLinear:
4. Termination Condition
   1. Does your training stop eventually?
   2. Does your training stop prematurely?
   3. Does your training stop when the training error is at the minimum?
   4. Does your training stop when the validation error is at the minimum?
   5. Do you have any other observations?

**Linear Dataset: Terminate only due to reaching 1000 total iterations, for 8 different Learning Rates**



