Summary:

* Supervised, non-parametric, and instance based algorithm
* The algorithm essentially boils down to forming a majority vote between the K most similar instances to a given “unseen” observation where similarity is defined according to some distance metric.
* These metrics could be Manhatttan, chebyshev, or hamming but in this instance we are using Euclidean distance.

Training:

* Nothing!
* KNN is a lazy algorithm that only memorizes the training data
* Since the data is already memorized i.e. stored in a list, we just return the training data passed, without manipulation.

Predictions:

* When the predictor receives a new observation
  + Needs to compute the Euclidean distance between the "new" observation and all the data points memorized
    - We store these distances in an array and then we sort it from nearest to farthest
  + Next the predictor selects the k nearest memorized observations based on distance
    - We store these in a list as well
  + The new observation is then assigned the label that is most common among those k memorized observations
    - We use the most\_common extension of the counter function in order to return the proper label

Comparison:

* As the chart indicates, sometimes SKLearn is more accurate and sometimes not, but on average they perform about the same
* The bigger exception to this though is that our implementation takes about five times longer than the SKLearn implementation
* On a small dataset like this is ok but would have negative impacts on processing bigger data