Given two points, **p** and **q**, of dimension (# features) *n*, belonging to dataset X,(pi (qi) being **p** (**q**) ‘s value mapped in the ith dimension):

If the *n* axes were imagined to form an *n*-dimensional grid, the Manhattan Distance (MnD) is the number of steps (taken in just one of *n* dimensions at each iteration) required to go from **p** to **q** in the *n*-dimensional grid. This ‘taxicab’ distance is formalized by MnD(**p**,**q**) = |p1 – q1| + |p2 – q2| + … +|pn – qn|. On the other hand, the Euclidean Distance (ED) measures the direct distance between **p** and **q** ‘as the crow flies’ - it is the length of a straight line connecting **p** and **q**, and does not involve keeping the path constant in all but 1 dimension over the course of ‘travel’; formalized by HD(**p,q**) = [(q1-p1)2 + (q2-p2)2 + … + (qn-pn)2]0.5. However, correlation between different dimensions (features) can preferentially skew the distribution of data points in our *n*-dimensional space. This will bias the absolute distances (MnD or ED) measured between p and q. We can counter this by comparing each point against the distribution of all members of X - thus, for a given point **p**, for each dimension *i,* we normalize pi based on the variance observed in our entire dataset for that dimension. Mahalanobis Distance allows us to do exactly this - measure the distance between **p** and **q**, using covariance between the dimensions as a normalization factors (since we know that **p,q** are both data points belonging to population X). This is formalized by D2=(**p**−**q**)TC−1(**p**−**q**) where C is the covariance matrix of X. When all covariances are null (all dimensions uncorrelated) and the variances of all dimensions are equal, MD = ED.

The edit distance (measure of transformations required to convert a string to another) is a suitable measure of distance between entities when the factors are categorizable (ex. characters in strings). Given two strings A, B of length *m*, *n* respectively, the Levenshtein distance (LvD) between them is the minimum number of character insertions (I), deletions (D), and substitutions (S) required to transform A into B. The ‘cost’ of each action (I, D, S) can be initialized to different values, with the sum of the entirety of actions required being the LvD(A,B). Hamming Distance (HD) provides the upper bound for possible LvD between 2 sequences, when *m*=*n*. It represents the minimum number of substitutions required to change A to B. Clearly, depending on the modifications we allow on the system, LvD (or HD) is preferable. When comparing strings of equal length, LvD will give us a lower cost for approximate string matching than would HD (consider ABCDE vs BCDEF : 1 D + 1 I in LvD, 5 S in HD).

Resources consulted:

An Introduction to Bioinformatics Algorithms (Computational Molecular Biology), Neil Jones and Pavel Pevzner, MIT Press, 2004

<http://blogs.sas.com/content/iml/2012/02/15/what-is-mahalanobis-distance.html>

<http://stats.stackexchange.com/questions/62092/bottom-to-top-explanation-of-the-mahalanobis-distance>

<https://en.wikipedia.org/wiki/Edit_distance>