Naïve Bayes algorithm

Input:

1. Data matrix of dimension n by m (n rows and m columns)
2. Training labels L. Each label l_i is an integer indicating the class that row i belongs to.

Algorithm:

1. Training:
    a. Initialize the mean m_j of each class to the vector of all 1's (not 0's). These are pseudocounts to avoid zero variance.
    b. Compute mean m_j of each class by adding each datapoint to m_j and dividing by number of datapoints
2. Prediction: Assign point x'_i to class j if x'_i is closest to the mean of class j normalized by standard deviation. In other words

$$class(x'_i) = argmin_j \sum_{k=1}^{m} \left(\frac{x'_{ik} - m_{jk}}{s_{jk}} \right)^2$$

where s_{jk} is the standard deviation of the k^{th} feature (dimension) of class j and m_{jk} and x'_{ik} are the k^{th} entry (dimension) of the vectors x'_i and m_j respectively.